

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



**Open-Domain Web-based Multiple Document
Question Answering for List Questions
with Support for Temporal Restrictors**

Patricia Nunes Gonçalves

DOUTORAMENTO EM INFORMÁTICA
ESPECIALIDADE CIÊNCIAS DA COMPUTAÇÃO

2015

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



**Open-Domain Web-based Multiple Document
Question Answering for List Questions
with Support for Temporal Restrictors**

Patricia Nunes Gonçalves

Tese orientada pelo Prof. Doutor António Horta Branco

Especialmente elaborada para obtenção do grau de
DOUTOR EM INFORMÁTICA
ESPECIALIDADE CIÊNCIAS DA COMPUTAÇÃO

2015

Dedico esta tese em memória do meu pai, meu grande ídolo.

Acknowledgements

Obrigada a todos que contribuíram diretamente e indiretamente na realização desse trabalho. Obrigada e meus colegas do NLX, especialmente para meu inseparável amigo João Silva por todo apoio e amizade nestes últimos anos.

Quero também agradecer a minha família pelo apoio incondicional, especialmente para minha mãe que foi uma rocha nesses últimos anos, meu irmão Guti, minha cunhada Grasi e minha sobrinha Duda pelo carinho e a todos meus tios, primos e ao meu padrinho Tio Pedro (in memoriam) pelo apoio e carinho.

Obrigada a todos amigos que fiz ao longo desses 5 anos, Marina, Branquinho, Renata, Marcia, Leandro, Maria Clara, Laly, Cecilia. Monica, Giu e Jojo, obrigada por me acolherem logo na minha chegada e por vários momentos felizes. Rosa e Augusto obrigada por me fazer apaixonar pelo tango e pelos jantares maravilhosos. Obrigada Lê e Vini que me “salvaram” no momento mais crítico da minha vida que foi no nascimento do meu filho. Obrigada a Luciana pelos nossos cafés inesquecíveis e a toda família “The Christians”, Chris, Zazá e Tico pelo carinho e amizade. A Dalva, Tácito e Samsam pelos maravilhosos passeios aos fins-de-semana. Ao Zeca e sua família, Juliana e Antonio por todo apoio e incentivo. A Cristiane, Marcirio e nosso príncipe Vicente meu agradecimento especial por me acompanharem nessa jornada com um apoio fundamental.

Quero agradecer a Deus por me trazer meu maior presente, meu filho Andrew and to his Daddy, Colin to make my dreams come true when I become a mom.

Obrigada a meu orientador António Branco por acreditar neste trabalho e um agradecimento especial aos professores Renata Vieira e Thiago Pardo que mesmo de longe me incentivaram a desenvolver esta tese.

Finalmente, agradeço à Fundação para Ciência e Tecnologia pelo financiamento desta tese de doutoramento através da bolsa SFRH/BD/65647/2009 e ao projeto QTLeap concedido pela European Commission sob o contrato #610516.

Abstract

With the growth of the Internet, more people are searching for information on the Web. The combination of web growth and improvements in Information Technology has reignited the interest in Question Answering (QA) systems. QA is a type of information retrieval combined with natural language processing techniques that aims at finding answers to natural language questions.

List questions have been widely studied in the QA field. These are questions that require a list of correct answers, making the task of correctly answering them more complex. In List questions, the answers may lie in the same document or spread over multiple documents. In the latter case, a QA system able to answer List questions has to deal with the fusion of partial answers. The current Question Answering state-of-the-art does not provide yet a good way to tackle this complex problem of collecting the exact answers from multiple documents.

Our goal is to provide better QA solutions to users, who desire direct answers, using approaches that deal with the complex problem of extracting answers found spread over several documents. The present dissertation address the problem of answering Open-domain List questions by exploring redundancy and combining it with heuristics to improve QA accuracy. Our approach uses the Web as information source, since it is several orders of magnitude larger than other document collections. Besides handling List questions, we develop an approach with special focus on questions that include temporal information. In this regard, the current work addresses a topic that was lacking specific research.

A additional purpose of this dissertation is to report on important results of the research combining Web-based QA, List QA and Temporal QA. Besides the evaluation of our approach itself we compare our system with other QA systems in order to assess its performance relative to the state-of-the-art. Finally, our approaches to answer List questions and List questions with temporal information are implemented into a fully-fledged Open-domain Web-based Question Answering System that provides answers retrieved from multiple documents.

Keywords: Question Answering, Web-based, List Question, Temporal List Question

Resumo - Portuguese Abstract

Com o crescimento da Internet cada vez mais pessoas buscam informações usando a Web. A combinação do crescimento da Internet com melhoramentos na Tecnologia da Informação traz como consequência o renovado interesse em Sistemas de Respostas a Perguntas (SRP). SRP combina técnicas de recuperação de informação com ferramentas de apoio à linguagem natural com o objetivo de encontrar respostas para perguntas em linguagem natural.

Perguntas do tipo lista têm sido largamente estudadas nesta área. Neste tipo de perguntas é esperada uma lista de respostas corretas, o que torna a tarefa de responder a perguntas do tipo lista ainda mais complexa. As respostas para este tipo de pergunta podem ser encontradas num único documento ou espalhados em múltiplos documentos. No último caso, um SRP deve estar preparado para lidar com a fusão de respostas parciais. Os SRP atuais ainda não providenciam uma boa forma de lidar com este complexo problema de coletar respostas de múltiplos documentos.

Nosso objetivo é prover melhores soluções para utilizadores que desejam buscar respostas diretas usando abordagens para extrair respostas de múltiplos documentos. Esta dissertação aborda o problema de responder a perguntas de domínio aberto explorando redundância combinada com heurísticas. Nossa abordagem usa a Internet como fonte de informação uma vez que a Web é a maior coleção de documentos da atualidade. Para além de responder a perguntas do tipo lista, nós desenvolvemos uma abordagem para responder a perguntas com restrição temporal. Neste sentido, o presente trabalho aborda este tema onde há pouca investigação específica.

Adicionalmente, esta dissertação tem o propósito de informar sobre resultados importantes desta pesquisa que combina várias áreas: SRP com base na Web, SRP especialmente desenvolvidos para responder perguntas do tipo lista e também com restrição temporal. Além da avaliação da nossa própria abordagem,

comparamos o nosso sistema com outros SRP, a fim de avaliar o seu desempenho em relação ao estado da arte. Por fim, as nossas abordagens para responder a perguntas do tipo lista e perguntas do tipo lista com informações temporais são implementadas em um Sistema online de Respostas a Perguntas de domínio aberto que funciona diretamente sob a Web e que fornece respostas extraídas de múltiplos documentos.

Palavras Chave: Sistema de Resposta à Perguntas, Domínio Aberto, Perguntas do Tipo Lista, Perguntas do Tipo Temporal

Contents

1	Introduction	1
1.1	Question Answering	3
1.2	Types of Questions	5
1.3	Major Topics	8
1.3.1	Open-Domain QA	8
1.3.2	Web-based QA	9
1.3.3	List QA	10
1.3.4	Temporal QA	11
1.4	Motivation and Goals	12
1.5	Challenges	14
1.6	Outline of this Dissertation	16
2	Related Work	19
2.1	Outline	19
2.2	Question Answering Evaluation Competitions	20
2.3	Evaluation on Question Answering	21
2.4	QA Systems for the Portuguese Language	24
2.5	Approaches for List Questions	27
2.6	Approaches for Web-based QA Systems	33
2.7	Approaches for Temporal QA Systems	38
2.8	Summary	43
3	Answering List Questions	45
3.1	Outline	46
3.2	List Questions	46
3.3	Challenges	49
3.4	Approach	50
3.4.1	Expected Input	51

CONTENTS

3.4.2	External Resources and Support Tools	52
3.5	Architecture	54
3.5.1	Question Processing	54
3.5.1.1	Question Analysis	55
3.5.1.2	Transformation of the Question into a Query	56
3.5.1.3	Keywords Extraction	56
3.5.1.4	Semantic Category of the Expected Answer	57
3.5.1.5	Counting Keywords	58
3.5.1.6	Identifying the Question-Focus	58
3.5.2	Passage Retrieval	59
3.5.2.1	Document Retrieval	59
3.5.2.2	Document Analysis	60
3.5.3	Answer Extraction	64
3.5.3.1	Extracting Candidate Answers	65
3.5.3.2	Building the Answer List	66
3.6	LX-ListQuestion: an Open-Domain Web-based QA system for List Questions	69
3.7	Summary	71
4	Answering Temporal List Questions	73
4.1	Outline	74
4.2	Background	74
4.2.1	Classification of Temporal Questions	76
4.2.2	Challenges in Temporal Question Answering	79
4.2.3	Related Work	79
4.3	Approach	80
4.3.1	Design Features	81
4.3.2	Expected Input	81
4.3.3	Classification of Temporal List Questions	82
4.3.4	Solving Time-Range	83
4.3.4.1	Temporal Restriction in an Absolute Time	83
4.3.4.2	Temporal Restriction with a Relative Reference	83
4.3.4.3	Temporal Restriction with a Vague Reference	84
4.4	Architecture	84

CONTENTS

4.4.1	Question Processing Module	85
4.4.1.1	Identifying the Temporal Expression	86
4.4.1.2	Defining the Temporal Boundaries	88
4.4.1.3	Generating the Query	89
4.4.2	Document Processing Module	89
4.4.3	Answer Processing Module	90
4.5	LX-ListQuestion: QA system to List Question with Temporal Restrictors .	92
4.6	Summary	93
5	Evaluation	95
5.1	Outline	96
5.2	Automatic Evaluation	96
5.3	Question Dataset	98
5.4	List Questions: Evaluation	101
5.4.1	The role of Document Retrieval	101
5.4.2	Evaluation using Different Threshold Parameters	102
5.4.3	Evaluation with Different Setups	103
5.5	Comparing LX-ListQuestion and other QA Systems	105
5.5.1	LX-ListQuestion versus RapPortagico	106
5.5.2	LX-ListQuestion versus XisQuê	108
5.5.3	LX-ListQuestion versus START	111
5.5.4	LX-ListQuestion versus WolframAlpha	114
5.6	Temporal List Questions: Evaluation	117
5.6.1	Evaluation for Different Temporal Restrictors	118
5.6.2	Evaluation of Temporal List Question over Questions from Págico	119
5.6.3	Evaluation of Temporal List Question over Questions from Yahoo!	
	Answers	120
5.7	Summary	122
6	Conclusions	125
6.1	Summary	125
6.2	Contributions	129
6.3	Future Research Directions	131

CONTENTS

A	Question Dataset	135
A.1	Págico Dataset	135
A.2	QALD Dataset	141
A.3	Temporal Dataset	142
B	LX-ListQuestion Answers	147
C	XisQuê Answers	169
D	START Answers	177
E	Wolfram Alpha Answers	183
	References	193

List of Figures

1.1	Google search engine.	1
1.2	Question answer system	2
1.3	Generic architecture for a question answering system	4
3.1	Question answering system architecture	54
3.2	Question processing module	55
3.3	Passage retrieval module	59
3.4	Before and after the segmenter tool.	60
3.5	Document analysis process	61
3.6	Web page title matches the question.	62
3.7	Sentence matches the question.	63
3.8	Answer extraction module.	64
3.9	Example: candidates extracted by LX-Ner	65
3.10	Example: candidates extracted by question-focus	66
3.11	Building the list of answers.	68
3.12	LX-ListQuestion - online version architecture.	69
3.13	LX-ListQuestion - online version interface - list answers.	70
3.14	LX-ListQuestion - online version interface - wordcloud Answers.	71
4.1	Example of TimeML annotation.	75
4.2	Summary of the processing of temporal list questions.	85
4.3	Document analysis for temporal list questions.	89
4.4	Extracting candidate answers for temporal list question.	90
4.5	Finding co-occurring temporal expressions.	91
4.6	LX-Listquestion online QA system - list question with temporal restrictors	92
5.1	Original answers given by Páxico competition.	96
5.2	Answers given by Páxico competition after cleaning.	97

LIST OF FIGURES

5.3	Variation in correct answers	97
5.4	Relaxed candidate matching in the automatic evaluation.	98
5.5	Correct answers found on the corpus.	101
5.6	Display of XisQuê system operation.	109
5.7	Display of START system operation.	111
5.8	Display of WolframAlpha system operation.	114
B.1	LX-ListQuestion QA system answering the question Pagico_004	147
B.2	LX-ListQuestion QA system answering the question Pagico_054	148
B.3	LX-ListQuestion QA system answering the question Pagico_062	149
B.4	LX-ListQuestion QA system answering the question Pagico_086	150
B.5	LX-ListQuestion QA system answering the question Pagico_088	150
B.6	LX-ListQuestion QA system answering the question Pagico_100	151
B.7	LX-ListQuestion QA system answering the question Pagico_109	151
B.8	LX-ListQuestion QA system answering the question Pagico_112	152
B.9	LX-ListQuestion QA system answering the question Pagico_133	153
B.10	LX-ListQuestion QA system answering the question Pagico_140	153
B.11	LX-ListQuestion QA system answering the question QALD_010	154
B.12	LX-ListQuestion QA system answering the question QALD_028	155
B.13	LX-ListQuestion QA system answering the question QALD_032	156
B.14	LX-ListQuestion QA system answering the question QALD_036	157
B.15	LX-ListQuestion QA system answering the question QALD_062	158
B.16	LX-ListQuestion QA system answering the question QALD_074	159
B.17	LX-ListQuestion QA system answering the question QALD_114	160
B.18	LX-ListQuestion QA system answering the question QALD_141	161
B.19	LX-ListQuestion QA system answering the question QALD_155	162
B.20	LX-ListQuestion QA system answering the question QALD_176	163
B.21	LX-ListQuestion QA system answering the question TP_001	164
B.22	LX-ListQuestion QA system answering the question TP_002	165
B.23	LX-ListQuestion QA system answering the question TP_003	166
B.24	LX-ListQuestion QA system answering the question TP_004	166
B.25	LX-ListQuestion QA system answering the question TP_005	167
C.1	XisQuê QA system answering the question Pagico_004	169

LIST OF FIGURES

C.2	XisQuê QA system answering the question Pagico_054	170
C.3	XisQuê QA system answering the question Pagico_062	170
C.4	XisQuê QA system answering the question Pagico_086	171
C.5	XisQuê QA system answering the question Pagico_088	171
C.6	XisQuê QA system answering the question Pagico_100	172
C.7	XisQuê QA system answering the question Pagico_109	172
C.8	XisQuê QA system answering the question Pagico_112	173
C.9	XisQuê QA system answering the question Pagico_133	173
C.10	XisQuê QA system answering the question Pagico_140	174
C.11	XisQuê QA system answering the question TP_001	174
C.12	XisQuê QA system answering the question TP_002	175
C.13	XisQuê QA system answering the question TP_003	175
C.14	XisQuê QA system answering the question TP_004	176
C.15	XisQuê QA system answering the question TP_005	176
D.1	START QA system answering the question QALD_010	177
D.2	START QA system answering the question QALD_028	178
D.3	START QA system answering the question QALD_032	178
D.4	START QA system answering the question QALD_036	178
D.5	START QA system answering the question QALD_062	179
D.6	START QA system answering the question QALD_074	179
D.7	START QA system answering the question QALD_114	180
D.8	START QA system answering the question QALD_141	180
D.9	START QA system answering the question QALD_155	181
D.10	START QA system answering the question QALD_176	181
E.1	Wolfram Alpha QA system answering the question QALD_010	183
E.2	Wolfram Alpha QA system answering the question QALD_028	184
E.3	Wolfram Alpha QA system answering the question QALD_032	185
E.4	Wolfram Alpha QA system answering the question QALD_036	186
E.5	Wolfram Alpha QA system answering the question QALD_062	187
E.6	Wolfram Alpha QA system answering the question QALD_074	188
E.7	Wolfram Alpha QA system answering the question QALD_114	189
E.8	Wolfram Alpha QA system answering the question QALD_141	190

LIST OF FIGURES

E.9	Wolfram Alpha QA system answering the question QALD_155	190
E.10	Wolfram Alpha QA system answering the question QALD_176	191

List of Tables

1.1	Examples of temporal expressions and their temporal constraint	12
2.1	Subjective assessment.	24
2.2	Portuguese QA summary	27
2.3	List QA summary	32
2.4	Web-based QA - summary	37
2.5	Temporal QA summary	42
3.1	Examples of list questions	46
3.2	Examples of simple and complex list questions	47
3.3	Examples of list questions with constraints.	47
3.4	Examples of list questions with question-focus.	48
3.5	Examples of answering list questions	48
3.6	Examples of list questions expected input of LX-ListQuestion.	52
3.7	Examples of different type of list questions.	55
3.8	Semantic category of the expected answer	58
3.9	Examples of counting keywords	58
3.10	Examples of question-focus identification	58
3.11	Sentence classification.	64
4.1	The relations defined by Allen (1983)	74
4.2	Tags of TimeML.	75
4.3	Examples of expected input of LX-ListQuestion.	81
4.4	Examples of questions with temporal restriction in an absolute time.	83
4.5	Examples of questions with temporal restriction with a relative reference	84
4.6	Examples of questions with temporal restriction with a vague reference	84
4.7	Examples of patterns to identify temporal restriction in an absolute time	86
4.8	Examples of patterns to identify temporal restriction with a relative reference	87

LIST OF TABLES

4.9	Examples of temporal questions without explicit datetime mark.	88
4.10	Examples of mapping of the expression to a time interval.	88
4.11	Examples of questions and query	89
4.12	Building the answer list for temporal list questions.	91
5.1	Subset of question dataset - Págico Competition	100
5.2	Subset of question dataset - QALD	100
5.3	Corpus composition	102
5.4	Evaluation using different threshold parameters.	103
5.5	Evaluation of the LX-ListQuestion system	104
5.6	Comparing QA systems - RapPortagico and LX-ListQuestion	106
5.7	Evaluation of QA systems - RapPortagico and LX-ListQuestion	106
5.8	Comparing answers - LX-ListQuestion and RapPortagico	107
5.9	Comparing QA systems - XisQuê and LX-ListQuestion	108
5.10	Evaluation of QA systems - XisQuê and LX-ListQuestion	110
5.11	Comparing answers - XisQuê and LX-ListQuestion	110
5.12	Comparing QA systems - START and LX-ListQuestion	112
5.13	Evaluation of QA systems - LX-ListQuestion and START	113
5.14	Comparing answers - LX-ListQuestion and START	113
5.15	Comparing QA systems - WolframAlpha and LX-ListQuestion	115
5.16	Evaluation of QA systems - LX-ListQuestion and WolframAlpha	116
5.17	Comparing answers - LX-ListQuestion and WolframAlpha	116
5.18	Evaluation for different temporal restrictors.	118
5.19	A set of question dataset of Págico Competition.	119
5.20	Evaluation of temporal list questions: LX-ListQuestion versus RapPortagico.	120
5.21	Question coverage: LX-ListQuestion versus RapPortagico.	120
5.22	Temporal questions given by Yahoo!Answers.	121
5.23	Evaluation of temporal list questions: LX-ListQuestion versus XisQuê.	121
5.24	Question coverage: LX-ListQuestion versus XisQuê.	122
6.1	Results overview	133

1

Introduction

The quantity of readily available electronic information is growing every day. With the development of the Internet, more people are searching for information on the Web and more tools are able to explore large amounts of text.

Information Retrieval (IR) techniques are used to return relevant documents in response to a user query. Unfortunately, in their current stage of technological maturity, IR systems do not directly provide the appropriate information. Instead, the users have to extract answers from documents that are returned by IR systems.

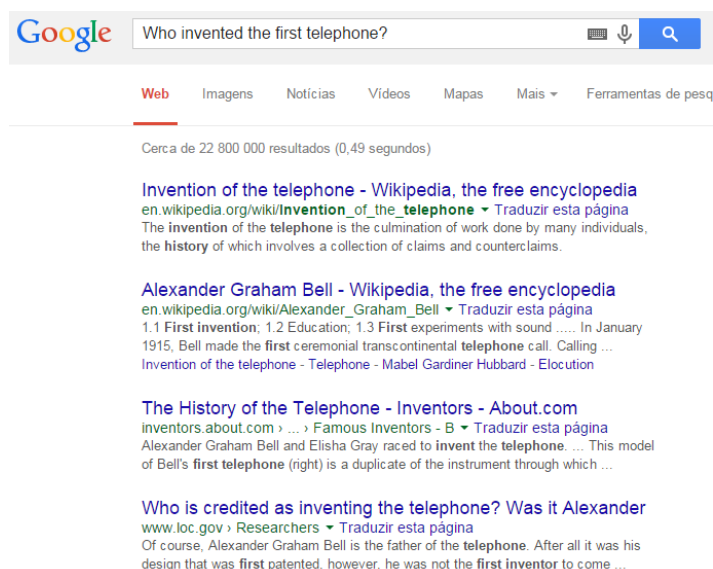


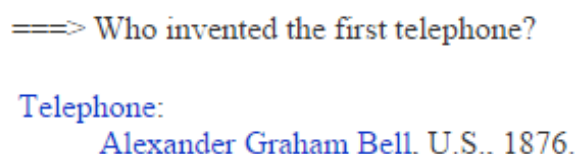
Figure 1.1: Google search engine.

1. INTRODUCTION

Figure 1.1 shows an example of the result of searching for “Who invented the first telephone?” in the search engine Google, which has become synonymous of web search. The answer to this question appears in the piece of information (snippet) provided by the 4th link returned. Despite the development in recent years, these systems do not provide exact answers, and the user needs to read the documents or read all pieces of information retrieved by the search engine to locate the desired answer.

Located at the intersection of Information Retrieval and Natural Language Processing (NLP), Question Answering (QA) is a challenging task involving the extraction of relevant answers to natural language questions from large text collections (Paşca, 2003). In contradistinction to IR systems, in a QA system the user can search for information by writing correct interrogative sentences instead of a few keywords. QA becomes *information* retrieval, as originally intended, where the right answer is extracted from documents and not just some links to relevant documents.

Figure 1.2 shows the same example of an user searching for “Who invented the first telephone?” in a QA system, which provides an exact answer to the user.



==> Who invented the first telephone?

Telephone:
Alexander Graham Bell, U.S., 1876.

Figure 1.2: Question answer system

Chapter Outline

In this introductory Chapter, we start with Section 1.1, presenting the background in the Question Answering area. Section 1.2 details the type of questions currently studied in the literature. In Section 1.3 we present the major topics for this doctoral research: Open-Domain QA, Web-based QA, List QA and Temporal QA. The motivation and main goals are presented in Section 1.4. Finally, in Section 1.5 we present the challenges that are addressed in this research.

1.1 Question Answering

Research on QA started around the 1960's with the development of Baseball (Green *et al.*, 1961), a system that answered questions about baseball games. The input question is analyzed and the information requested is extracted from the data stored in list structures. Some years later, Lunar (Woods *et al.*, 1974) and Qualm (Lehnert, 1978) appeared. Lunar was a QA system for lunar geologists. Qualm was a QA system that uses the theory of conceptual information processing based on models of human memory organization.

In the 1980's and early 1990's, large digital repositories of data were put in place, which coupled with more powerful computers paved the way for empirical, data-driven and robust approaches to QA.

Currently, QA systems have been combining elaborate natural language processing techniques, linguistic representation and machine learning methods to make the processing of textual information progress over the years. The QA system process that provides a precise answer is quite different from the task of Information Retrieval or Information Extraction, but it depends on both of them as important components (Strzalkowski and Harabagiu, 2007).

A QA System can be classified in terms of: Information Source, Domain, Language, Data Pre-Processing, Approach Complexity, Answer Rendering, User Interface and Interactivity.

- **Information Source:** (i) Structure: The information source can be: unstructured (plain text) or structured (XML, Database); (ii) Type: The data can be of different types: texts, images, maps, audio, etc; (iii) Data quality: high quality (books) or poor quality (Internet); (iv) Size: large corpus or small corpus; (v) Form: static (corpus) or dynamic (Internet) information source.
- **Domain:** (i) Closed: domain specific (e.g. medicine, biology, law); (ii) Open: domain independent.
- **Language:** (i) Monolingual: QA systems in one language only; (ii) Multilingual: QA systems in several languages.
- **Data Pre-Processing:** (i) pre-processing: at the time of system execution, the source texts are already pre-processed and annotated with some kind of linguistic information, they are often also reviewed manually. (ii) without pre-processing: the texts are processed at the time of system execution.

1. INTRODUCTION

- **Approach Complexity:** (i) Shallow methods: use local features for the processing, such as predefined patterns, template matching, string similarity and others. (ii) Deep methods: use more sophisticated linguistic processing to extract the answer, such as, deep parsers, logical forms, etc.
- **Answer Rendering:** (i) Extracted: Answers are extracted verbatim from the information source; (ii) Generated: Answers are generated on the basis of source text using techniques based on text entailment, templates, etc.
- **User Interface:** (i) QA Systems based on written questions; (ii) QA Systems based on spoken questions.
- **Interactivity:** (i) Interactive: QA Systems that are able to request additional clarification information from the user (similar to a dialogue); (ii) One-way: QA Systems that answers the question input by the user without any further interaction with the user.

The generic architecture for a Question Answering System consists of three modules: question processing, passage retrieval and answer extraction. The generic architecture is presented in Figure 1.3

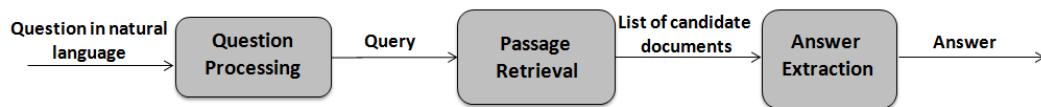


Figure 1.3: Generic architecture for a question answering system

Question Processing Module: Question Processing is responsible for transforming the question into a format that can be processed by a query engine, by determining relevant keywords present in the input question. This module is also responsible for recognizing the types of questions and the types of answers. This information is an important component to help the system to deliver the right answer at the end of the processing.

Passage Retrieval Module: The Passage Retrieval module is responsible for selecting relevant passages or documents. Its main task is the control of the search space. If the number of passages is too small, the query may be resubmitted to the query engine using more general terms. Otherwise, if the number of passages is very large, the query may be resubmitted using more specific terms. This module is also responsible for removing redundant or irrelevant information.

Answer Extraction Module: Answer extraction is responsible for choosing the most accurate answer. The retrieved passages are ranked and the possible answers are isolated. Factual QA systems usually provide 1 to 5 possible answers. The whole answer is composed by short-answer and justification. The short-answer is the most accurate answer and the justification is the text that supports the answer. Usually, the short-answer is extracted from the justification and it provides the context for the answer.

1.2 Types of Questions

The types of question currently studied in the literature are: Factoid, Definition, Complex, Boolean, List and Temporal.

So called factoid questions are the most studied and several QA approaches have been developed for them over the years. These questions, which syntactically are categorized as Partial Interrogative clauses, usually start with an interrogative pronoun. The pronouns commonly used are: *Who, What, Where, Which, When, How many/much*. Factoid questions can also start with a verb. From a syntactic point of view, sentences that start with verbs are called Imperative clauses. These are commonly used in QA systems to obtain answers to subjacent questions. The most recurrent verbs used are the following: *name, say, indicate, mention, determine, specify, disclose, report, check*, among others.

(1) Examples of Factoid Question:

Where is the residence of the prime minister of Spain?

Who is the mayor of Tel Aviv?

What is the area code of Berlin?

How many countries are there in Europe?

Name one investment higher than the one made by APDL in the last twenty years.

1. INTRODUCTION

Even though factoid questions have been studied since 90's of last century, the results obtained still leave room for improvement. Factual questions are becoming increasingly complex and traditional approaches cannot answer such questions satisfactorily. There is a large effort in the area to improve the results to factoid questions.

Definition questions, usually start with *What is*. The research on the handling of definition goes beyond Question Answering and has spawned its own field. The definition can: (i) provide a description of the concept; (ii) indicate a synonym relationship; (iii) describe the function of the concept or (iv) enumerate the part of the concept.

(2) **Examples of Definition Question:**

What is an atom?

What is FTP?

What is an Ontology?

So called complex questions usually start with the interrogative pronoun *How* or *Why*. The answer to this kind of questions is extremely hard to find because that generally involves semantic processing. There is an effort in the area to develop systems capable of answering such type of question.

(3) **Examples of Complex Question:**

How will Luís Camões Square be adorned?

Why is Itamar criticizing the Brazilian Government?

Boolean questions, from a syntactic perspective, are Total Interrogatives sentences. They typically elicit a confirmation or a denial as an answer. In Boolean questions the answer is already explicit in the question and the interrogative word does not appear in the question. Total Interrogatives can be subdivided into three types: affirmative, negative or alternative.

(4) **Examples of Boolean Question:**

Is proinsulin a protein?

Didn't Italy make it to the EURO quarterfinals?

Can the exam be done with a declaration form his embassy or with a residence certificate?

List questions is another type of questions now starting to be widely studied. For these questions, it is expected not a single answer but a list of answers. In a certain sense, List questions can be viewed as a complex version of factoid questions because they have some similar characteristics, namely they can start with a pronoun or a verb. The complexity of List questions lies in the search for the answer but also in determining the number of instances that the answer requires.

In List questions, the answer may lie in the same document; when the answer is already a list, e.g., list of cities in Portugal: Lisbon, Coimbra, Porto e Faro; or the answer is spread in multiple documents; e.g. (document A) : “Lisbon is the capital of Portugal.” (document B:) “Porto is a very important city in Portugal.”, etc. In this case, a QA system able to answer List questions has to deal with the fusion of partial answers.

(5) **Examples of List Question:**

Which countries have the white, green and red colors in their national flag?
What European Union countries have national parks in the Alps?
Name rare diseases with dedicated research centers in Europe.

Despite Temporal questions being factoid questions, a new field has emerged to specifically study this type of questions. Authors define Temporal questions differently. In this dissertation we assume three different types of Temporal questions: (i) Simple temporal question are questions that require a date as an answer; (ii) Complex temporal question are questions with a temporal expression that contains more than one event related with temporal sign (e.g. temporal sign: *before*, *after*, *when*, etc.); (iii) Temporally restricted question are factoid questions that require a answer restricted to a time interval. We discuss Temporal Questions in further detail in Chapter 4.

(6) **Examples of Temporal Question:**

When did Christopher Reeve die?
What happened to world oil prices after the Iraq annexation of Kuwait?
Which city hosted the World Cup in 1994?

1. INTRODUCTION

1.3 Major Topics

In this Section we will present the major topics for this doctoral research: Open-Domain QA, Web-based QA, List QA and Temporal QA.

1.3.1 Open-Domain QA

In Open-domain Question Answering the range of possible questions is not constrained, hence a much tougher challenge is placed on systems. The goal of an Open-domain QA system is to answer questions on any kind of subject domain. Research in Open-domain Question Answering began in 1999 with the Text REtrieval Conference (TREC), which provides large-scale evaluation of QA systems thus defining the direction of research in the QA field.

The evaluations initially focused on factoid questions, which remain until now the main focus of interest for the development of Open-domain QA systems. However, throughout the years, the question dataset proposed by TREC has become more complex, increasing the difficulty level of the competition. In 2003, list and definition question were included in addition to factoid questions.

In the following years, constraints were added to the factoid questions, making the questions more complex. The most common constraints are: Temporal, Geographic and Quantitative. Temporal constraints are related to time (months, years, centuries and so on), e.g. “What are the Brazilian poets who published volumes with ballads **until 1941?**” Geographic constraints are related to localization (cities, region, countries, continents and so on), e.g. “What are the rare diseases with dedicated research centers **in Europe?**” Quantitative constraints are related with how often something happens, e.g. “What Belgians won the Ronde Van Vlaanderen **exactly twice?**”

Research on Open-domain Question Answering is not limited to the complexity of the questions but also encompasses the complexity of the document collection. Initially the document collection of TREC was composed of articles from newspapers, then Web pages and blog articles were added to the document collection thus increasing the complexity of the development of QA systems. Mining blogs and Web pages for answers introduce new challenges in the Open-domain field requiring (i) the development of systems that handle texts that are not well formed, and (ii) dealing with discourse structures that are more informal and less reliable than newswire.

1.3.2 Web-based QA

Web documents have been used as corpora for many tasks in NLP and can also be used as an information source for QA systems. The World Wide Web is an extremely large repository of publicly accessible documents covering any topic, making it an obvious source to obtain information.

A Web-based QA system differs from the others in many ways, the most important being the use of a search engine to retrieve Web pages that potentially contain answers to the question. Search engines provide a convenient front-end for accessing and filtering enormous amounts of Web data.

Another distinguishing feature of Web-based QA systems is the choice of the type of data to work with. Some systems choose to work only with the snippet (text fragments retrieved by the search engine) thus reducing the search space, as is the case of Lamp (Zhang and Lee, 2003), AskMSR (Zhang and Lee, 2003) and Qualim (Kaisser and Becker, 2004). Others choose to work with the full document retrieved from the link provided by the search engine, such as Mulder (Kwok *et al.*, 2001), NSIR (Radev *et al.*, 2002) and AnswerBus (Zheng, 2002).

In addition to viewing the Web as a repository of unstructured information, some researchers also take advantage of semi-structured information contained in the HTML pages. Cucerzan and Agichtein (2005) use the information found in HTML tables which typically summarizes various interesting relations. Other semi-structured data in the HTML pages that can be explored are: itemized lists, meta-data information (usually contains keywords of the content), titles with explicit markup in HTML, among others.

Recently, approaches have appeared that work with structured data using knowledge databases freely available on the Web. In this case, the system does not use a search engine but a structured query language named SPARQL¹ that allows searching for information in the Web in a structured way. An example of a system that uses this approach is Deanna (Yahya *et al.*, 2013b) which uses a database named YAGO².

While the Web is undeniably a useful resource for Question Answering, it is not without drawbacks (Guda *et al.*, 2011). Useful information on the Web is often drowned out by the

¹ See more in <http://www.w3.org/TR/rdf-sparql-query/> - Last access on December, 20 - 2014.

² See more in <http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/> - last access in December, 20 - 2014.

1. INTRODUCTION

amount of irrelevant information. The major issue in the use of the Web as a information source lies in developing an approach capable of separating right from wrong answers.

1.3.3 List QA

List questions started being studied in the context of QA in 2001 when TREC included this type of questions in the dataset. The first List questions in the question dataset of TREC explicit regarding the number of items required in the list of answers. Thus, determining the number of items in the answer of List questions was not an issue to be considered by the QA systems at that time.

(7) **Examples of List questions with the number of items required explicit in the question:**

- a. What are 9 novels written by John Updike?
- b. What are 12 types of clams?
- c. Name 8 Chuck Berry songs.

Later, the number of items in the answers was removed from the questions which increased the complexity of processing. The systems, at the time were not prepared to tackle this complex problem, that is to find the optimal size of the list for each question. One way to handle this issue was to make the QA system return always a fixed number of responses (usually between 10 and 20 elements) for all List questions.

(8) **Examples of List questions where the number of items required is not explicit in the question:**

- a. Which airlines use Dulles Airport?
- b. Name books written by C.S. Lewis.
- c. Name all Chuck Berry songs.

Until now, finding the number of correct items in the list of answers remains the key challenge of List questions QA.

Finding the correct answers to List questions requires discovering a set of different answers in a single document or across several documents. An approach to answer a List question in a single document is very similar to the approach to find the correct answer to factoid

questions: (i) find the most relevant document; (ii) find the most relevant excerpt, usually by searching for textual patterns and (iii) extract the answers from this relevant excerpt.

(9) **Examples of sentences that contain a list of answers:**

- a. The list of planets are: Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus and Neptune.
- b. Some examples of Beatles songs: “All You Need Is Love”, “Here Comes the Sun”, “Yellow Submarine”.
- c. Ringo, Lennon and McCartney are members of the Beatles.

When the answers that are extracted are spread over several documents many new challenges are raised, such as grouping repeated elements, handling more information, separating the relevant information from the rest of the information, among others. This dissertation addresses this challenging task by extracting and rendering answers from multiple document retrieved from the Web.

1.3.4 Temporal QA

Temporal QA requires the extraction of temporal information encoded in natural language text (Schilder and Habel, 2003). Temporal processing requires the recognition and processing of temporal information given by the question. A questions can refer directly or indirectly to a temporal expression:

- Direct temporal expressions are the most common in the questions. In such cases, the temporal expression is explicit in the question, e.g. What country controlled Syria in **1930**?

Paşca (2008) uses hand-build patterns to identify facts associated with a temporal expression and builds a repository with information retrieved from the Web. The most recent work of Yahya *et al.* (2012) uses a knowledge database to answer questions with temporal restrictions.

- Indirect temporal expressions, also named Complex Temporal Questions by some authors, are more interesting and have been particularly studied over recent years, e.g.

1. INTRODUCTION

What did George Bush do after the U.N. Security Council ordered a global embargo on trade with Iraq?.

Harabagiu and Bejan (2005) developed templates associated with the complex temporal question and use inference to find the answer in the annotated corpus. Tao *et al.* (2010) explores an OWL ontology over clinical data to find events related with time about the patients. The system handles complex temporal questions with this approach.

After the identification of the temporal expression in the question, it is necessary to transform the temporal expression into temporal constraints. The temporal constraint is interconnected with the time-range of the temporal expression. Temporal processing is essential in QA to successfully address a time constraint in the question. Examples of temporal expression and their temporal constraint may be found on Table 1.1:

Temporal Expression	Temporal Constraint
in 1967	1967 - 1967
between 1950 and 1965	1950 - 1965
after 1980	1980 - current date
before 1970	[...] - 1969
in the 20th century	1901 - 2000

Table 1.1: Examples of temporal expressions and their temporal constraint

1.4 Motivation and Goals

With the growth of the Internet, more people are searching for information on the Web. Typically, the user enters a set of keywords into the search engine and gets a list of links to web pages as a result. The task of finding the desired answer among the results that were returned then falls on the user.

This process is the same when the user searches for list answers. Consider this scenario: the user wishes to find a list of European countries. To do this, the user inserts a few keywords into a search engine, for instance, *European countries*, and is quite likely to find a web page containing the desired information among the first hits returned by the search engine. In other words, when the information that is needed is trivial, and a web page with the full answer already exists, a search engine may cope with this task.

However, when the information that is needed is non-trivial and it is found spread over several documents, a lot of human effort is required to gather the various separate pieces of data into the desired result, which is not an easy task. The current state-of-the-art, be it in IR or QA, does not provide a good way yet to tackle this complex problem.

Users do not generally want to go through several documents and put a lot of effort in finding the desired answer. Ideally, they would prefer to quickly get a precise answer and go on to make use of it instead of spending time searching and compiling the answer from pieces spread over several documents. Our motivation is to provide better QA solutions to users who desire direct answers to their queries, by investigating approaches for dealing with the complex problem of extracting answers found spread over several documents and use them to compile as complete and correct list of answers as possible.

In addition to dealing with the problem of answering List questions, we want to develop an approach that focus on answering Temporal List questions. In order to find answers that satisfy the temporal restriction in the question, our approach needs to ensure that the temporal expression is identified and the temporal constraint correctly solved.

Our research is guided by the following property of QA over free text captured dynamically in the Web: Answers may appear redundantly in many places and in many forms.

Our key goals are:

- ***Investigate appropriate ways of processing and rendering answers spread over multiple documents exploiting the redundancy of information available in the Web to improve accuracy of List QA;***
- ***Develop an approach of Shallow Temporal Processing to answer temporal list questions in order to improve the state-of-the-art.***

To achieve these goals we divide them into the following specific sub-objectives:

- Perform an in-depth study of approaches for Open-Domain Questions Answering Systems for large text collections that provide answers to natural language questions that require list of answers.
- Study techniques that allow creating the list of answers from the items that are found spread over multiple documents while coping with redundancy, aiming to improve the state-of-the-art.

1. INTRODUCTION

- Ensure that the approach is able to process questions of multiples types: (i) interrogative sentences; (ii) imperative sentences or (iii) keyword-based queries.
- Build a system in such way that it does not require pre-indexing of documents, allowing it to provide answers in real time and make sure that the system can handle noisy and unstructured data, thus allowing it to run directly over the Web.
- Review the approaches in the current state-of-the-art in Temporal QA field.
- Gather, through a study of corpora, the most common temporal expressions in the List questions and extend the QA system with a shallow temporal processing module to find the list of answer to temporal list questions aiming to improve the state-of-the-art.
- Finally, implement our approach to answer list questions and temporal list questions as a fully-fledged on-line QA system that provides an as much accurate and complete list of answers extracted from multiple documents as possible.

1.5 Challenges

There have been many advances in the area of QA. With the increasing complexity of questions being considered, the development of a QA system continues to be a challenging task. Furthermore, the challenges go beyond the development of the QA system itself.

We split the challenges into four groups: (1) general challenges about the QA area; (2) issues specific to the development of a Open-domain Web-based QA System; (3) issues specific to the development of a QA System for List Questions; (4) issues specific to the development of a QA System for Temporal List Questions.

General challenges in QA:

- Natural language: Working with natural language is a challenge in itself. For instance, the linguistic diversity in writing varies greatly. Often, when developing a system, tests are run over a specific domain, e.g. academic thesis. When one then tries to run the system over another linguistic domain or genre, e.g. newspaper corpus, there is usually a drop in performance.

- Dependency on the performance of NLP tools: despite the effort in the area, the NLP tools available still have much room of improvement. If a QA system is based on a set of NLP tools, the QA system performance is tied to the performance of those NLP tools.
- Redundant data: QA system must identify the same element even if it is written in a different way. For example, when written in another language, *Florence- Firenze* or when abbreviated, *IBM - International Business Machine Corporation*.

Challenges specific to Web-based QA System:

- Noisy data: There is no control over the textual quality of the documents extracted from the Web. The problem is to separate relevant from irrelevant information. A sentence having a high score can indicate that the answer is probably there. However, it may be a false positive and not have any relevance to answering the question.
- Unstructured documents: Beyond the problem of textual quality, the extracting of documents from the Web raises another issue, namely the problem of unstructured data. Furthermore, not all of the documents retrieved from the Web (usually HTML) are well formed. The challenge is that the QA system should deal with the problem of data that are not always in the HTML standard format.
- Dynamic extraction: Extracting Web documents dynamically faces many problems such as broken links, diversity of the returned files (eg .PDF, .PS). A robust QA system must be prepared to deal with these issues.
- Processing requirements: The processing requirements of an automatic QA system running over the Web are quite heavy. Nonetheless, the system must return an answer to the user in a timely manner. If the system takes a long time to provide an answer, the user may not use it.
- Evaluation: This is a critical topic issue given that when building up a corpus dynamically from the Web, one is never sure that the documents retrieved have a answer for the question being asked. Hence, the evaluation of a system QA over the web should be made in a qualitative manner. In particular, when a QA system gives a wrong answer one should evaluate whether the set of documents retrieved contained the answer

1. INTRODUCTION

or not. If one does not do this type of review, it is not determined whether the fault is in the collection of documents or in the QA system itself.

Challenges specific to List Questions:

- List accuracy: The challenge is to identify accurately whether if each of the elements belong or not to the final list of answers, i.e. to ensure that the list is as accurate as possible and that there are no extraneous elements.
- List completeness: The QA system must ensure that the quantity and the quality of documents retrieved are enough to search for these elements to ensure that all relevant elements are present in the final list of answers.

Challenges specific to Temporal List Questions:

- Identifying temporal expression: The challenge is identify the temporal expression and define the correct temporal constraint.
- Accurate answers: A QA system needs to ensure that the list of answers satisfies the temporal constraint given by the question.

1.6 Outline of this Dissertation

This dissertation is organized as follows. Chapter 2 presents an overview of the state-of-the-art of the main topics of this dissertation: Question Answering for Portuguese language, QA systems that handle List questions, Web-based QA systems and Temporal QA system.

Chapter 3 covers one of the major goals of this dissertation, developing a Web-based QA system for List questions. First we provide an overview of List questions and we remind the key challenges to this dissertation. Our overall approach to answer List questions is discussed and the design features are presented. We also describe in detail the architecture and modules of the system, and present examples. Finally, we present the user interface of our Open-domain Web-based QA system.

Chapter 4 provides the background for Temporal QA and presents the main concepts and challenges. We present the design features of LX-ListQuestion, the questions that are

expected as input and we explain how we solve the time-range issue of the temporal expression. We present in detail the design features of our Temporal Processing Module and describe the architecture and modules of the system.

Chapter 5 is divided into three parts. In the first part we will explain the evaluation of LX-ListQuestion in different ways: (i) we present all correct answers found in corpora of different size; (ii) afterwards, we test the system using different threshold parameters and (iii) we present the evaluation of LX-ListQuestion using four different setups in order to test the efficiency of our approach. The second part compares the results against four different QA Systems. In order to assess the positioning of our system in the state-of-the-art, our evaluation has two components: the quantitative evaluation of answers and the question coverage evaluation. In the final part of this Chapter, the Temporal Processing module of LX-ListQuestion is evaluated. We test our approach by applying different temporal restrictors for the same base question and compare the results for Temporal List Questions against other QA systems.

In Chapter 6 we summarized all chapters in this dissertation and a number of contributions to the research in the field of List QA and Temporal QA are present. We also present the future directions of this research.

2

Related Work

This research work aims to develop a fully-fledged Web-based QA system to answer natural language questions that require answers in the form of Lists extracted from multiple documents retrieved from the Web of Portuguese pages. Besides handling List questions, the system focus on questions that include a temporal restriction. In order to achieve this goal, we begin with a review of the current state-of-the-art of Question Answering Systems in relevant areas: (1) QA systems for the Portuguese language, (2) QA systems that handle List questions, (3) Web-based QA systems and (4) QA systems that specialize on Temporal Questions.

2.1 Outline

This Chapter presenting a background over QA Competitions and Measures. Different evaluation competitions have been organized over the years which are presented in Section 2.2. While Section 2.3 presents various evaluations measures for QA systems that have been used.

In Section 2.4 we present QA systems developed for the Portuguese Language. Most of these systems were developed to participate in the Cross-Language Evaluation Forum (CLEF), which often require a pre-processing step in which the corpus is pre-processed (e.g. POS tagging, named entity annotation, etc.) and stored in a way that facilitates search (e.g. indexing by keywords). At last we present the XisQuê, a QA system for Portuguese that answers factoid questions from the open set of documents from the Web.

2. RELATED WORK

Section 2.5 is dedicated to discussing approaches that focus on answering list questions. Statistical and machine learning approaches have been used to handle list questions. Other approaches that exploit Semantic Content from Wikipedia are also frequently used.

Recently, research has been devoted to the task of developing open-domain QA systems based on collections of real world documents using Web as a corpus. Section 2.6 presents the different approaches that aim to find correct answers in the Web. Most systems answer only factoid questions. There is a great variety in the approaches that are used to handle Web corpora, such as redundancy, probabilistic models, clustering, etc.

QA systems that specialize on temporal questions are also presented in this chapter. In Section 2.7 we address this issue and present the approaches developed to solve simple and complex temporal questions. The features common to these systems are the restriction to closed-domain, the reliance on pre-tagged corpora and the usage of knowledge databases.

2.2 Question Answering Evaluation Competitions

Different evaluation competitions have been organized over the years. They are organized with the objective of promoting the progress of QA systems. Examples of competitions are: TREC¹, CLEF², QALD³, GikiCLEF⁴ and Páxico⁵.

TREC is an annual event sponsored by the U.S. National Institute of Standards and Technology (NIST) which has been held annually since 1999. The TREC goals are: (i) to encourage research on information retrieval; (ii) to exchange resource ideas between industry, academia and government; (iii) to demonstrate improvements in retrieval methodologies and (iv) to make available evaluation techniques in information retrieval.

The Cross-Language Evaluation Forum (CLEF) promotes R&D in multilingual information access by (i) developing an infrastructure for the testing, tuning and evaluation of information retrieval systems operating on European languages in both monolingual and cross-language contexts, and (ii) creating test-suites of reusable data which can be employed by system developers for benchmarking purposes. CLEF has been held annually since 2003.

¹trec.nist.gov

²www.clef-campaign.org

³ <http://www.sc.cit-ec.uni-bielefeld.de/qald>

⁴www.linguateca.pt/GikiCLEF

⁵www.linguateca.pt/Pagico

QALD is a series of evaluation campaigns on multilingual question answering over linked data, currently part of the Question Answering lab at CLEF that started in 2011 and it is held annually since then. The motivation of QALD is based on lots of structured data published on the Web and how typical users can access this knowledge becomes of crucial importance. The key challenge in QALD is to translate the users information needs into a form such that they can be evaluated using standard Semantic Web query processing and inferencing techniques.

GikiCLEF was an evaluation competition specifically designed to investigate cultural and linguistic issues. GikiCLEF was organized under the scope of CLEF 2009 containing 50 topics in Bulgarian, Dutch, English, German, Italian, Norwegian, Portuguese, Romanian and Spanish. The information source was the Wikipedia collections in all these languages. The topics required a list of answers. GikiCLEF has been held only once in 2009.

Págico was an evaluation competition that aims to encourage the development of QA system for Portuguese which it has had a single edition in 2012. The Págico Corpus is composed by Portuguese Wikipedia pages. The question dataset contains 150 topics among major subjects: Letters, Arts, Geography, Culture, Politics, Sports, Science and Economy. These topics were determined with special attention to ensure that the search for the answers is non-trivial, since the topics require multiple answers which are to be found spread throughout multiple documents.

2.3 Evaluation on Question Answering

In this section we introduce the evaluations measures for QA systems (Teufel, 2007). The metrics most commonly used are: recall, precision and F-measure. When we use these metrics we must take into consideration two lists: a reference list (correct answers expected) and the system list (answers returned by the QA system). These metrics are described below:

- **Precision:** C is the number of common elements between the reference and the system lists and S is the number of elements given by the system.

$$precision = \frac{C}{S}$$

2. RELATED WORK

- **Recall:** C is the number of common elements between the reference and the system lists and L is the number of elements in reference list.

$$recall = \frac{C}{L}$$

- **F-measure:** Harmonic mean of precision and recall. F_1 gives equal weights to precision and recall, which simplifies to:

$$F_1 = \frac{2*recall*precision}{recall+precision}$$

Other measures commonly used in the competitions like TREC or CLEF are: Accuracy, Mean Reciprocal Rank and Hit Success:

- **Accuracy**, the fraction of questions judged to have at least one correct answer in the first n answers to a question. Let $C_{D,q}$ be the correct answers for question q known to be contained in the document collection D and $F_{D,q,n}^S$ be the first n answers found by system S for question q from D then accuracy is defined as :

$$accuracy^S(Q, D, n) = \frac{|\{q \in Q \mid F_{D,q,n}^S \cap C_{D,q} \neq \emptyset\}|}{|Q|}$$

- **Mean Reciprocal Rank (MRR)** gives information about the capacity of a system to return a correct answer in the first result. The Mean Reciprocal Rank can be computed as:

$$MRR = \frac{\sum_{i=1}^{N_q} \frac{1}{rank_i}}{N_q}$$

Where N_q is the number of questions and $rank_i$ is the rank of the first correct answer.

- **Hit Success:** gives a measure of ability of the system to provide a correct answer in the top n answers. It is usually calculated for the first answer ($n = 1$) or among the top-3 answers ($n = 3$). The Hit Success is computed as:

$$top - n = \frac{\#CR_{i \leq n}}{\#questions}$$

Where $\#CR_{i \leq n}$ is the number of correct answers in ranks between 1 and n, the $\#questions$ is the number of questions.

Since MRR and Hit Success do not apply to List questions, the evaluation metrics that will be used in this work are: recall, precision and f-measure.

Human Evaluation

For the evaluation of QA Systems to a factoid questions, a type of subjective assessment is also used. This considers the relation between short-answer and justification and document retrieval. Table 2.1 shows a summary of subjective assessment. For each answer the assessor gives one of these 6 scores:

- Full: the elements are provided by: (1) the precise short-answer, (2) the document justifying this answer and (3) a correct textual passage extracted from the text.
- Right: the precise short-answer extracted from the document justifying this answer, but the justification is not provided.
- Unsupported: the precise short-answer is given but but the document from which is was extracted does not justify it.
- Supported: the short-answer does not provides the right answer to the question but the passage is relevant.
- Inexact: the short-answer does not provides a right answer to the question but it is extracted from a document justifying it.
- False: wrong answer, document and textual passage.

2. RELATED WORK

Assessment	Short-answer		Justification	Right Document
	Exact	Correct		
Full	YES	-	YES	YES
Right	YES	-	NO	YES
Unsupported	YES	-	NO	NO
Supported	NO	-	YES	YES
Inexact	-	YES	NO	YES
False	NO	-	NO	NO

Table 2.1: Subjective assessment.

2.4 QA Systems for the Portuguese Language

In this section we discuss the state-of-the-art QA for Portuguese language. We present different approaches developed in the literature.

Senso Question Answering System (Saías and Quaresma, 2009) participated in CLEF Campaigns in 2004, 2005 and 2008. The system uses texts that were pre-processed with a syntactical parser and indexed for future search. It has three main modules: The local KB, query and solver. Local KB contains common sense facts about places, entities and events. The query module is responsible for the analysis of the question and for selecting a set of relevant documents. The solver module search for the answer by using two approaches: ad-hoc solver (answer detection that can be directly detected in the text) and logic solver (a logic programming based tool that searches for answer being aware of the semantic expressed in local KB). The system performed with an accuracy of 46.5% for all questions. The same techniques are used for list questions. The question-dataset of CLEF 2008 have 10 list questions and the system answered correctly 3 of them and performed with 33.33% accuracy.

Esfinge (Costa, 2005, 2006) is a general domain Portuguese QA System which has participating in CLEF since 2004. Each text in the collection of documents was divided in sets of three sentences and these sentences were pre-processed using NLP Tools. The system uses external tools like a syntactic analyzer, a morphological analyzer and a named entity recognizer. Each type of question has patterns related to it that will be used to retrieve relevant text passages, for example, Question: Who is Stephen Hawking?, Pattern: Stephen Hawking is. These patterns are searched in the sentences collections pre-processed and also

2.4 QA Systems for the Portuguese Language

in the Web using the Yahoo¹ web search. The system performed with an accuracy of 23.5% (first answer) and 30.5% (all answers) at CLEF 2008.

The QA@L2F System (Coheur *et al.*, 2009) participated in CLEF 2007 and 2008. The system implements 3 main steps. In the corpus pre-processing step, relevant information is extracted, like named entities and the relations between them, which is then stored in a database. Question Interpretation transforms the question into a frame. Each frame contains question type, question target, name entities and auxiliaries (like verbs, adjectives and adverbs). Answer Extraction uses three approaches: linguistic reordering, named entity matching and brute force plus NLP. The system performed with 20% of accuracy in CLEF 2008.

IdSay (Carvalho *et al.*, 2009) is an Open Domain Answering System for Portuguese. The system participated at CLEF 2008 and reports an accuracy of 32.5% (first answer) and 42.5% (all answers). The module of question analysis uses specific patterns to identify the question category. The document retrieval module searches in previously indexed documents for a set of texts that contains all words and entities present in the question. The passage retrieval module selects passages with less than 60 words. The experiment was repeated in Carvalho *et al.* (2010) with a new version of IdSay. The authors identified strengths and weakness when the system was compared to others QA Systems for Portuguese. They improve the system using semantic information based on Wikipedia² and TeP (Thesaurus for Portuguese)³. From Wikipedia they extract entity synonyms and filters based on Ontological knowledge. The new IdSay improves the accuracy of 32.5% to 50.5% without degradation of response time. IdSay does not have any special treatment for list questions.

The approach of Raposa QA System (Sarmiento, 2006; Sarmiento *et al.*, 2008a) is based on Named Entity Recognition as the main strategy to answer *Factoid* and *Definition* questions. The system uses a set of 25 rules to analyze the type of question and identify the answer type. The document collection are stored in a database. They choose some cue words like family relations, locations and proper names to make the index in the database. The selection of answer is based on redundancy. Raposa system performed with 14.5% of accuracy at CLEF 2008.

The Priberam QA System (Amaral *et al.*, 2009, 2006; Cassan *et al.*, 2006) uses indexation of documents. The indexation is an off-line procedure by which a set of target documents is

¹ Available on <http://www.yahoo.com/> - Last access on July, 15 2014.

² Available on <http://www.wikipedia.org/> - Last access on July, 15 2014.

³ Available on <http://www.nilc.icmc.usp.br/tep2/> - Last access on July, 15 2014.

2. RELATED WORK

parsed in order to collect information in index files. The system analyzes the question and classifies the question into categories (86 types) and defines the question pattern (QP). Each question pattern has a question-answer pattern (QAP) associated. The system also extracts pivot, which are key elements of the question like words, expressions, name entity, phrases, numbers, dates and abbreviations. Document retrieval submits a query to the index files using as search keys the pivot lemmas, their heads of derivation and their synonyms. Each pivot and synonym has a weight associated. The system will select the sentence based on this weight and a score is calculated. The right answer will be extract if the sentence matches the QAP. The system performed with 63.5% of accuracy at CLEF 2008.

RapPortagico is a QA framework developed by Rodrigues and Oliveira (2012) which participated in Páxico Competition in 2012. The main approach combines the indexing of the content in the Wikipedia pages and identifying the noun phrases in the questions. The approach takes advantage of synonyms using lexical ontology to expand the questions. The official results in Páxico Competition (Mota, 2012) reports a F-measure of 0.104 to RapPortagico, which was the system with best performance in the competition.

The main features that sets XisQuê (Branco *et al.*, 2008a) (Branco *et al.*, 2008b) (Rodrigues, 2007) apart from the other systems are: real-time system, web-based and open-domain. XisQuê receives a question in Portuguese and the process is on-the-fly without any pre-processing for indexing of documents. The answers are searched in and extracted from the web. The input questions may address issues from any subject domain like Sports, History, etc. The system handles “Who”, “When”, “Where” and “Which-X” type of questions. For the answer extraction, the system matches the sentences selected with linguistic patterns. The overall MRR value obtained for XisQuê is 0.73 when short and long answer are considered and 0.48 when only short-answers are taken into account.

Summary

This section presented an overview of QA system for Portuguese. The majority of these systems was developed with a focus on the CLEF campaign and use document pre-processing and indexing as a starting point, such as Saías and Quaresma (2009), Costa (2005), Coheur *et al.* (2009), Carvalho *et al.* (2009), Sarmiento *et al.* (2008a) and Amaral *et al.* (2009).

We found only one system that fully retrieves the information from the Web and processes it in real-time, XisQuê (Branco *et al.*, 2008a). Our system also works with texts from the Web. Up to now, to our knowledge, no QA system that specifically answers List questions from the Web texts was developed for the Portuguese language. Table 2.2 shows the approaches of QA system for Portuguese language in a summarized form.

System	List Question	Corpus	Pre-Indexing	Approach
Saias and Quaresma (2009)	No	CLEF + Web	Yes	Semantic
Costa (2006)	No	CLEF + Web	Yes	Patterns
Coheur <i>et al.</i> (2009)	No	CLEF	Yes	NLP Tools
Carvalho <i>et al.</i> (2009)	No	CLEF	Yes	String Matching
Sarmiento <i>et al.</i> (2008b)	No	CLEF	Yes	Redundancy
Amaral <i>et al.</i> (2009)	No	CLEF	Yes	Patterns
Branco <i>et al.</i> (2008a)	No	Web	No	Patterns
<i>LX-ListQuestion</i>	<i>Yes</i>	<i>Web</i>	No	<i>Redundancy + Heuristics</i>

Table 2.2: Portuguese QA summary

2.5 Approaches for List Questions

List questions differ from factoid questions because they expect not a single item answer but a list of answers. The complexity of List questions lies not only in the search for the answer but also in determining the number of instances that the answer requires. Several approaches have been used to answer List questions. The most common approach is to take a QA system for factoid questions and extend it to answer List questions.

The Factoid QA system returns only one answer while for List questions the system returns N answers, where this value N is assigned differently by each system. Usually the value 10 is assigned for N. Some systems using this approach are Gaizauskas *et al.* (2005) and Wu and Strzalkowski (2006). The performance of these systems is very low, with less than 0.10 f-score value. In what follows, we present other major approaches pursued in List questions.

2. RELATED WORK

Exploiting NLP Tools and Linguistic Resources

Answering questions using several NLP tools and linguistic resources has been investigated by some researchers like Hickl *et al.* (2006) Yang *et al.* (2003). These NLP tools and resources include named entity recognition, PropBank, NomBank, FrameNet, semantic dependency, coreference resolution, WordNet, ontology and others.

LCC Chaucer-2 (Hickl *et al.*, 2007) is a QA system developed for combining several strategies for modeling the target of a series of questions and optimizing the extraction of answers. The system works for factoid questions and List questions. Chaucer uses five different answer extraction strategies: (i) Entity-Based (when the question is about two named entities); (ii) pattern-based (hand-crafted patterns); (iii) soft pattern-based (automatically generated); (iv) FrameNet-based (tries to match the answer against a semantic frame dependencies); (v) Predicative Question-Based (similarity metrics). They relied on the same techniques used for basic factoid questions to answer List questions. Chaucer performed with an accuracy of 56.1% for factoid questions and 32.4% F-Score for List questions on TREC 2007.

QUALIFIER - Question Answering by Lexical Fabric and External Resources (Yang *et al.*, 2003) is developed using a modular platform and for each type of question a system module is responsible for generating the response. The system handles *Factoid*, *Definition* and *List* questions. For *Definition* questions it uses techniques from information retrieval and summarization. For factoid questions it uses focus on events and uses external knowledge acquisition to relate named entities and events. From this information, rules are extracted and the factoid questions are answered based on this rules. For List questions they seeking for cue words in the texts like: “*list of...*”, “*following list...*” or patterns like: < *same_type_NE* >, < *same_type_NE* > and < *same_type_NE* > + verb. They reports accuracy of 0.56 in factoid questions and 0.31 F-score in list questions over TREC-12 question dataset.

Statistical Approaches and Machine Learning

Statistical approaches are another approach to answering questions. The system developed by Whittaker *et al.* (2006) uses a statistical model based on the probability of an answer depending on a question. The system was developed to answer *Factoid*, *List* and *Definition* questions. The best result was 0.03 on precision score in List questions on CLEF 2006 question-dataset.

Zhou *et al.* (2006) uses a statistical approach to rank documents and sentences. They extract answers from the most highly ranked sentences and also extract answers from the 2 previous and the 2 next sentences with respect to the top ranked answers. During the extraction they calculate the distance score associated with each answer. The answer ranking procedure is a simple linear combination of three scores: document score, sentence score and maximum distance score. The system performed 0.165 F-score in List questions in TREC-15 question dataset.

Machine learning has also been used in the context of Question Answering. Fada (Find All Distinct Answers) system (Wang *et al.*, 2008; Yang and Chua, 2004a) employs the Web to support List questions answering. The system focuses on improving Document Retrieval Module with the technique of web page classification to find complete and distinct answers. The technique needs a large set of documents (around 3000 web pages per question), which are classified into 4 classes: Collection Page, Topic Page, Relevant Page and Irrelevant Page. The Decision Tree classifiers were trained using 29 features. It is important to highlight that this technique only has an impact when the complete list of answers is spread over different texts. They improve the results of 0.319 to 0.464 F-Score with this technique over the question-dataset of TREC 2003.

Razmara and Kosseim (2008) answer List questions using a clustering method to group candidate answers that co-occur more often in the collection. The answer type of questions is divided into nine classes: person, country, organization, job, movie, nationality, city, state and other. The documents are retrieved and all terms that match an answer type are extracted. The similarity is computed for each pair of candidate answers based on their co-occurrence within sentences. Having clustered the candidates and determined the most likely cluster, the final candidate answers are selected. The system achieved 0.163 F-score using the question dataset of TREC 2007.

Exploiting Relation between Answers and Questions

The approach based on relation between answers and questions are mostly exploited for factoid questions, and can be used for List questions as well. This approach uses information about Expect Answer Type (EAT) extracted from questions and builds patterns to map candidate answers in the texts. The candidate answers need to match with the EAT. Webber *et al.* (2002) mention other relationships like equivalency (mutually entailment), specificity (one-way entailment), alternativity and aggregation.

2. RELATED WORK

LiQED System (Kor, 2005) uses question-answer relation pairs as a base to create the main algorithm to find list answers. The system only answers List questions. A previous version of the system of Ahn *et al.* (2005) had a collection of patterns that identifies the answer for a factoid question and the top 10 answers compose the list. The new version of the system improves answer precision and recall of List questions. The new version uses these patterns of the old version to find sentences collection. The collection of sentences will be used to identify terms that are in some sense relevant to the question and its answers. The system uses the relevant terms to build new text patterns to find more relevant answers to compose the list. Even though the system uses this technique, they were unable to significantly improve the results. The system performance was 0.21 F-score in the previous version and 0.22 F-score in the new version over question dataset of TREC 2004.

Dalmas and Webber (2007) also uses relation between answer and question approach and developed two strategies: (i) baseline, scoring candidates according to their relation with question; (ii) fusion, strategy using the baseline score plus a score based on relationships between the candidates themselves. The relationship between answers is mapped using synonymy and hyperonymy WordNet¹ relation. This approach only works if the word is in WordNet and is coupled with a relation. No evaluation is presented.

Exploiting Semantic Content

Alternatively, some researchers propose to reformulate Question Answering with semantic content approach to answer complex questions that required multiple answers. This approach is based on answers that can be inferred using knowledge database to overcome the limitations of textual QA approaches (Cardoso *et al.*, 2010). Below we present some systems that adopted this approach.

The prototype system of Cardoso *et al.* (2009) processed the Portuguese Wikipedia and stored the information in a structured way. The system has 2 main modules: question interpreter and question reasoner. The question interpreter converts a natural language question into a machine-interpretable object representing the question. This object is composed by a subject (entity that defines the type of expect answers) and conditions (list of criteria that filter the answers). The question reasoner will decide on the best strategy to get the answers.

¹WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets. Synsets are interlinked by means of conceptual-semantic and lexical relations. See more: <http://wordnet.princeton.edu>

The strategies consist on a pipeline of SPARQL¹ queries made to knowledge resources to obtain and validate answers and their justifications. The system uses a named entity recognizer system and a geographic ontology as external resources. The paper does not provide score for the performance of the prototype.

GIRSA-WP (Hartrumpf, 2008; Hartrumpf and Leveling, 2010) is a QA System developed for the German Language. They use a technique similar to Cardoso *et al.* (2009) where the German Wikipedia was processed and the information stored as structured data. They transform the natural language question into an abstract relation to search in the database structure. The system integrates two basic systems: a deep QA System and a GIR system (Geographic Information Retrieval). The QA methods were extended with an interesting method of question decomposition which transform a complex question into a less complex one. The method has 6 different decomposition heuristics: temporal, local, coordinated, meronymy, description and operational. The main question is decomposed into one or more sub-questions. The answers to the sub-questions are used to find the answer in the main question. Question decomposition is a powerful technique that can provide answers to a complex questions. The problem of this technique is that the success rate depends on the success of finding right answers to the sub-questions. If the system did not find any answer, or found a wrong answer, the error is propagated to the main question resulting in a cumulative error. The best run of GIRSA-WP in the GikiCLEF achieved 0.14 of precision.

EQUAL system (Dornescu, 2009) relies on structural information from Wikipedia to extract answers. The semantic QA architecture of EQUAL is a set of semantic constraints which are used to represent the possible interpretations of questions. The constraints are extracted from each question. The constraints are: (1) entity (named entity into the question), (2) relation (main verb into the question), (3) temporal (temporal expression, time intervals, etc), (4) property (numeric values), (5) geographic (country, cities, region, etc) and (6) introduction (interrogative pronouns, imperative verb and declarative constructions). When extracting answers the constraints are verified and the candidate entities which satisfy all the constraints are selected. The question processing uses the nodes, relations and properties from Wikipedia ontology. This system ranking first among 8 systems in GikiCLEF, answering 25 out of 50 questions and achieved 47% of precision and 32% of recall.

¹ Available on <http://sparql.org/> - Last access on July, 15 2014.

2. RELATED WORK

Summary

In this section we presented the background work on QA Systems that focus on List questions. Table 2.3 summarizes the approaches presented in this section. Some systems simply reapply their techniques for factoid questions to List questions, but they show very low performance Gaizauskas *et al.* (2005) and Wu and Strzalkowski (2006).

Other systems explore NLP tools and resources. This approach seems to have achieved better results. However the time required for processing is very high and the performance of these systems depend on the performance of the supporting NLP tools Hickl *et al.* (2006) and Yang *et al.* (2003).

Systems that take advantage of Semantic Content also achieved good results although all information should be previously stored in a database. This approach seems suitable to QA system that focus on a specific domain where the information source can be limited and more easily stored Cardoso *et al.* (2009), Hartrumpf and Leveling (2010) and Dornescu (2009).

Our overview shows that approaches with statistical models and machine learning are viable solutions for QA system that needs to handle wide and noisy information, such as the one that is found on the Web, to extract List answers Whittaker *et al.* (2006) and Yang and Chua (2004a).

System	Corpus	Language	Main Approach
Hickl <i>et al.</i> (2006)	TREC	English	NLP tools
Yang <i>et al.</i> (2003)	TREC	English	NLP tools
Whittaker <i>et al.</i> (2006)	TREC	English	Statistical
Yang and Chua (2004b)	TREC + Web	English	Machine Learning
Razmara and Kosseim (2008)	TREC + Web	English	Clustering
Kor (2005)	TREC	English	Patterns
Hartrumpf and Leveling (2010)	Wikipedia	German English	Semantic Content
Cardoso <i>et al.</i> (2009)	Wikipedia	Portuguese	Semantic Content
Dornescu (2009)	Wikipedia	English	Semantic Content
<i>LX-ListQuestion</i>	<i>Web</i>	<i>Portuguese</i>	<i>Redundancy + Heuristics</i>

Table 2.3: List QA summary

2.6 Approaches for Web-based QA Systems

When Question Answering Systems extend the data source to include information available from the Web, new opportunities and challenges arise. In this section we present related work on Web-based QA systems. In the beginning of this section we present the pioneer Web-based QA systems, which first proposed to search for answers using Web as a corpus. Between 2000 and 2005 many works used redundancy as their main approach. Finally, we present the most current approach to search for answers in the Web.

The pioneer systems

MURAX is the oldest QA system using the Web as an information source. Developed in 1993 by Kupiec (1993), the system answers questions using an on-line encyclopedia. The main approach uses lexico-syntactic patterns. Evaluation was based on 52 Trivial Pursuit questions. The rate of correct answers is 75% in the top-5 answers.

MIT developed the START QA System by Katz (1997) in 1993 and it is still available on the Web¹. In Katz *et al.* (2003) a unified framework incorporating external knowledge sources in the QA process. The most prominent external source is the Wikipedia which is used as source to synonymy and hypernym information. To answer factoid questions, they use the snippet retrieved from Google search engine and heuristics based on matching for question focuses to select the right candidates. The process to answer List questions uses a knowledge base extracted from lists within in Wikipedia pages and also using heuristics based on matching for question focuses. The system achieves accuracy of 0.273 for factoid questions and F-score of 0.110 in TREC 2005.

Redundancy Approach

AnswerBus is an open-domain question answering system based on sentence level Web information retrieval (Zheng, 2002). The system accepts questions in different languages, such as German, French, Spanish, Italian and Portuguese. The questions are translated to English using BabelFish². The answer is always provided in English. The system uses the full documents to extract sentences that contain answers. The system scores the sentences

¹ Available on www.start.com - Last access on July, 12 2014.

² Available on <http://www.babelfish.com/> - Last access on July, 18 2014.

2. RELATED WORK

based on counting the number of keywords present in them. The performance was made over 200 factoid questions of TREC-8. The rate of correct answers is 70.5% in the top-5 answers.

The system presented in Clarke *et al.* (2001) exploits redundancy on the web to find answers. The success of this approach shows that the volume of information available in the Web is large enough to supply the answer to most factoid questions. To explore how redundancy can improve a QA system, the components of the QA system were simplified, e.g., the identification of candidates is made using a simple syntactic pattern that matches most names in English. The hypothesis was that redundancy could serve as a substitute for deeper analysis. For the experimentation was used a single category of question of TREC-9, namely those that require the name of a person as an answer (87 questions). The system answers correctly 49 questions (56% of them) and in 34 questions (39%) the correct answer was ranked in the first position.

AskMSR is a QA system developed by Microsoft Research Group (Banko *et al.*, 2002). The system takes advantage of data redundancy to find the correct answer. The main feature is the rewrite process by which a question is transformed into several queries, e.g., Question: “Where is the Louvre Museum located?”; query 1: “the Louvre Museum is located...”; query 2: “the Louvre Museum is in...”; query 3: “the Louvre Museum is near...”; and so on. The answers are selected by collecting 100 snippets per questions using the search engine. The system chooses 5 possible answers, ranks them, and presents the result to the user. The experiments used 500 factoid questions of TREC-9 and achieves the 0.507 in MRR metric (Dumais *et al.*, 2002).

Other Approaches: Clustering, Probabilistic and NLP Tools

Lamp is a web-based QA system developed by Zhang and Lee (2003) that takes advantage of the snippets in the search results returned for the Google search engine. The system answers only factoid questions and 100 snippets are retrieved by each question. The main algorithm is based on constructing a snippet cluster that contains the same plausible answers. The system was evaluated over 444 factoid questions of TREC-11 and achieves 0.47 on the MRR evaluation metric.

The system named Mulder (Kwok *et al.*, 2001), combines information retrieval with statistical natural language processing, lexical analysis and a voting procedure. The system ranks the candidates according to how close they are to keywords and clusters similar

answers together. The final answer is extracted from the highest scoring cluster. The experiment used 200 factoid questions of TREC-8 and the system answers correctly 34% of them.

Radev *et al.* (2002) introduce a Web-based question answering system, named NSIR, that uses a probabilistic method. Two probabilistic models were used, one based on N-gram and the other based on a vector space model. The system retrieved 40 documents per question using the Google search engine. The evaluation used 200 factoid questions from TREC-8. The system answers correctly to 164 questions. No precision results were reported.

QuASM is a QA system that exploits HTML structure (e.g. HTML tables and titles) of web pages to find the correct answer developed by Pinto *et al.* (2002). Using the HTML structure, the information is indexed into smaller units. No search engine was used, the system uses a specific crawler named fedstats¹ to retrieve the web pages. The crawler selects a collection of 177,670 documents. From these, a random set of documents was selected. These documents were used to generate 73 questions used in testing. The questions covered data in text tables, html tables and prose. These questions were used in the experiments; the authors report that the system answers 23 of them, though no recall or precision are presented.

Qualim is a QA system that uses an approach that rephrases the questions to find correct answers, e.g., from the question “*When did Amtrak begin operations?*”, the system generates the pattern “*Amtrak began operations in*”; and searches for matching strings in snippets returned by the search engine. Kaisser and Becker (2004) used the same technique to answer list questions and definition. The results reported are 0.343 for factoid, 0.125 for list questions and 0.211 for definition on TREC-2004.

Magnini *et al.* (2001) present DIOGENE Multilingual Web-based QA System. The system uses linguistic processing as main approach to answer factoid questions in Italian and English languages. The processing strongly relies on MULTIWORDNET for multiword expression recognition, word sense disambiguation and answer type identification. The search engine used was PRISE, developed by NIST². The system participated on TREC-11 main task and correctly answered just 10% of the questions. A manual analysis of a set of questions was made and the authors report the error analysis. Each module of the system was evaluated separately. The search engine produced a very high error rate of 53%, which means

¹ Available on <http://www.fedstats.gov> - Last Access on July, 21 2014.

² Available on <http://www.nist.gov/> - Last Access on July, 22 2014.

2. RELATED WORK

that the search engine retrieved a document containing the correctly answer for less than half of the questions. The named entity recognizer used also produced a high error rate of around 60%, mainly due to the low homogeneity between training and test corpus for this tool.

Recent Approaches

The proposal of Lloret *et al.* (2011) is to use text summarization to improve a factoid QA system. The system uses summaries instead of snippets to find the answer. The summarizer is integrated with the information retrieval stage. The Web pages are collected and summarized to extract the answer. The main approach uses textual entailment to remove redundancy of summaries and term frequency to score sentences. The main goal is to show the benefit of using summaries instead of snippets. The authors built their own question set consisting of 100 questions in English: 25 questions about person, 25 questions about organization, 25 questions about location, and 25 temporal questions. The system achieves 60.5% F-measure extracting answers from the summaries against 53.9% when using the snippets.

Wu and Marian (2011) propose a framework to aggregate query results from different sources in order to save users the hassle of checking query-related web sites to corroborate answers. The goal is to provide the best answers to the users using an individual score for each answer. The score takes into account the occurrence number, relevancy and originality of the sources reporting the answer, the prominence of the answer within the sources. When the scoring process is completed, the scores of the similar answers are aggregated. The system was evaluated over 142 factoid questions from TREC-9 and achieved 0.767 MRR score.

Deanna – Deep Answers for maNy Naturally Asked questions, is a QA system that comprises a suite of components for question decomposition, mapping components into the semantic concept space and generating alternative candidate mappings to a database. The system of Yahya *et al.* (2013b) uses SPARQL to search for answers directly in linked data sources like YAGO¹. The main focus of this work is in transforming the question into a correctly structured SPARQL query. The evaluation presented in Yahya *et al.* (2013a) reports the results obtained using QALD-2 question dataset. The system performed 0.45 F-measure for List questions and F-measure of 0.68 in Factoid questions.

¹ Available on <http://yago-knowledge.org> - Last Access July, 29, 2014

2.6 Approaches for Web-based QA Systems

Summary

Table 2.4 shows the overview of the state-of-art for Web-based QA system in a summarized form.

System	Information Source	Main Approach	Question Type	Available
Kupiec (1993)	Online Encyclopedia	Lexico-patterns	Factoid	No
Katz (1997)	External Knowledge Source	Redundancy, Patterns and Question focus	Factoid Definition List	Yes
Zheng (2002)	Web (full documents)	Keywords Frequency	Factoid	No
Clarke <i>et al.</i> (2001)	Web	Redundancy		No
Banko <i>et al.</i> (2002)	Web (snippets)	Rewritten query	Factoid	No
Zhang and Lee (2003)	Web (snippets)	Cluster	Factoid	No
Kwok <i>et al.</i> (2001)	Web (full documents)	Clustering Voting Procedure		No
Radev <i>et al.</i> (2002)	Web (full documents)	Probabilistic	Factoid	No
Pinto <i>et al.</i> (2002)	Fedestats website	HTML Structure	Factoid	No
Kaisser and Becker (2004)	Web (snippets)	Patterns	Factoid Definition List	No
Magnini <i>et al.</i> (2001)	Web	NLP (MultiWordNet)	Factoid	No
Lloret <i>et al.</i> (2011)	Web (full documents)	Summaries	Factoid	No
Wu and Marian (2011)	Web (full documents)	Score based on occurrence number, etc	Factoid	No
Yahya <i>et al.</i> (2013b)	Linked Data Source (YAGO)	Question transformation on SPARQL query	Factoid	No
<i>LX-ListQuestion</i>	Web (full documents)	Redundancy +Heuristics	List	Yes

Table 2.4: Web-based QA - summary

Research on Web-based QA system has been longstanding. We note that there is a lot of effort in creating systems capable of answering questions using the information available on the Web. Most systems answers factoid questions. The systems such as Katz *et al.* (2005) and Kaisser and Becker (2004) attempt to answers List questions. START by Katz *et al.* (2005)

2. RELATED WORK

uses a knowledge source extracted from lists in the Web pages to answer List questions. Qualim by Kaisser and Becker (2004) handles List questions using the Internet, although it does not have special treatment for List questions and uses the same technique to answer factoid and list question. Generally speaking the results in List questions using the Web as a information source is a challenging task, these systems achieved F-score of 0.110 reported by Katz *et al.* (2005) and 0.125 achieved by Qualim Kaisser and Becker (2004).

We note that some systems use the full documents and other snippets to find the correct answer. This decision has great impact when developing a Web-based QA system. Using full documents gives a greater chance of finding the answer, but it leads to a more time-consuming process because the texts are longer. Using snippets is the flip-side to this, providing faster processing due to smaller texts, but a lower chance of finding the answer in the returned snippet.

LX-ListQuestion aims to answer List questions using the Internet and so we chose to process the full documents to better select the most relevant information in the search for answers. This way we ensure that there is a greater chance of finding all the elements of the list of answers.

2.7 Approaches for Temporal QA Systems

The research on Temporal Question Answering has been active for nearly 15 years. Our study shows that TERQAS¹ was the starting point for research on temporal processing in QA. The workshop, held in 2002, address the problem of how to answer temporally based questions about the events and entities in text. This has led to a growing movement to build corpora annotated with temporal information that goes beyond simple times and dates, including events and temporal relations between events.

After TERQAS

Radev and Sundheim (2002) developed a corpus of temporal questions composed by 50 questions. The corpus was annotated manually with different information, e.g., the number of temporal expressions in the question, how many events are mentioned and their types, etc.

¹Workshop on Temporal and Event Recognition for Question Answering Systems. See more information on <http://www.timeml.org/site/tergas> - Last access on July, 24 2014.

This corpus was built to investigate how TimeML¹ can improve the performance of NSIR QA System (Radev *et al.*, 2002).

Not only corpora were developed to help the research in this area. A tagger capable of automatically annotating temporal expression was developed by Schilder and Habel (2003). The temporal tagger is capable of automatically annotating: (1) event information; (2) temporal information and (3) temporal relations (e.g. before, after, starts, finishes, etc.). The main goal is to determine if annotating the data with temporal information helps the temporal QA systems find the correct answer.

Using the Internet as an information resource for Temporal QA

The internet has been used to build data collections and repositories especially to answer temporal questions. Ahn *et al.* (2006) proposes two strategies to build a data collection for a temporal QA system using information available on the Web. Both strategies use hand-build patterns. The first strategy extracts and stores events from Wikipedia using shallow semantic interpretation, extracting events descriptions including temporal location and participants. The information stored is composed by event, date and description. Around 33,000 events are stored between 1600 and 2005 (about 19,000 are birth and death events). The second strategy searches the Web for temporal relations between events already extract from Wikipedia while applying the first strategy. Patterns are sent to the Google search engine and the snippets indicating temporal events between events are extracted, e.g., “<event1> gave way to <event2>”; “<event2> took place after <event1>”; “<event1> took place during <event2>”. The data collection was built to supporting temporal question answering systems.

In Paşca (2008), the authors describe how to build a repository using the nuggets (a sentence fragment retrieved by a search engine) to capture events and organize them into fact repositories. The focus is on answering temporal questions that require a date as an answer, e.g. “*When was the transistor invented?*”. The repository was built using lexico-syntactic patterns, e.g., “[Date] [when] [nugget]”; “[nugget] [in/on] [Date]”; “[verb] [optional ad-verb][in/on][Date]”. A fact retrieval framework was built to have immediate applications in web search, providing direct results for queries asking about a date or an event. Experimental evaluation was based on a quality assessment that analyzed feedback from 20 users. In their feedback, users included mostly positive comments about how the system performed.

¹TimeML is an annotation scheme for marking up events and time expressions and links between them. See more information on <http://www.timeml.org> - last access on July, 28 2014.

2. RELATED WORK

Some users felt that it was sometimes necessary to read the full-length document to assess the correctness of the dates. No evaluation about precision and recall were presented.

The framework developed by Tao *et al.* (2010) explores the semantic web as an environment to represent and reason about the temporal dimension of clinical data. An OWL ontology was used to build an API capable of finding events related with time. This API can return information about the duration of a given event, temporal relations between two events, the timeline of a set of events, among other information. The framework was built to search temporal information in the OWL ontology and allows the user to find concepts and relations with temporal information. The user can use historical information to infer new information from clinical narratives; e.g.; *Day 1: Patient's exams is normal; Day 2: Patient has body aches; Day 3: Patient starts medication; "Question: Did the patient body aches before starts with medication?"* To answer this question the system needs infer about the facts and time related to find the correct answers: Yes, the patient had body aches in the day 2 and starts medication on day 3. No evaluation about the performance of the framework are presented.

Another framework, named DEANNA, was built using structured knowledge bases available on the Internet (Yahya *et al.*, 2012). The system translates questions into a SPARQL query and searches for answers in databases such as DBPedia. In Yahya *et al.* (2013b) the issue of disambiguating temporal phrases in the question into temporal entities like dates, events and temporal predicates was addressed. In the knowledge base, time is associated with an event. Events are expressed through an event entity (e.g, World War) or an instance relation (e.g, was born in). The system can search the answers specifying the time window. The authors did not report those results for temporal questions.

Complex Temporal Questions

Temporal question answering offers a lot of interesting challenges. Some researchers develop methodologies and algorithms to temporal inference that aim to answer complex temporal questions. Harabagiu and Bejan (2005) introduce a methodology to temporal inference to use in QA systems. Temporal questions require several different forms of inference. The methodology uses annotation produced by TimeML to generate a template to answer temporal questions, named temporal signatures. When the temporal signatures are combined with sources of semantic inference and information about corefering events produce

sophisticated temporal inferences are produced. The temporal signatures are used to identify exact answers to temporal questions.

The work in Schockaert *et al.* (2006) present an algorithm based on an algebra to deduce temporal knowledge to answer temporal questions. Their focus is on complex temporal questions in which the question has a relation between two events, e.g., “Which battles were fought in Belgium between D-Day and the unconditional surrender of Germany?”. The authors use the data collection developed by Ahn *et al.* (2006) to support this approach. Temporal reasoning is further complicated by the fact that many historical events are vague. Their time span cannot be accurately captured by well-defined boundaries. No evaluation is presented.

Moldovan *et al.* (2005) discuss how temporal context reasoning can boost the performance of a QA system. The approach detects temporally related events in texts and converts them into a logical representation. The reasoning module uses a first order logic theorem prover. This reasoning module translate sentences, mark-up with temporal events, to a temporally enhanced first order logic assertion, this mean, transform the sentence in to a predicate, e.g., Predicate: “*Sentence1 contain Event1 and Event2*” from this predicate, the structure of logic representation extracted, e.g., Logic representation: *during(Event1,Event2)*. The predicates are generated based on hand coded interpretation rules. The main goal is to measure the contribution of temporal context to a QA system. The system achieved 37.1% correct answers with context against 29.1% without context.

The work presented in Saquete *et al.* (2009) is worth highlighting, in what regards the handling of complex temporal questions. The design of the system has a specialized layer to process complex temporal questions. The authors consider a taxonomy of temporal questions divided into four types of complexity: (1) Questions that require a temporal expression as an answer and do not contain temporal expression, e.g. “*When did man arrive on the moon?*” ; (2) questions that require a temporal reasoning of the temporal expression contained in the question, e.g. “*Who won the world cup 2010?*”; (3) Questions with a temporal expressions that contain more than one event related with temporal signal (temporal sign can be expressions like “before”, “after”, “between”, “when”), e.g. “*What did George Bush do after the U.N. Security Council ordered a global embargo on trade with Iraq in August 90?*” and (4) Questions without a temporal expressions that contain more than one event related with temporal signal, e.g. “*Who was the president of US when the AARP was founded?*”. The approach is based on decomposing the question into simple questions, according to the

2. RELATED WORK

temporal relations expressed in the original question. At the end of processing, the answers are recomposed to find the correct answers. An extensive evaluation is reported on Temporal Questions for English and Spanish languages. The evaluation was based on 200 temporal questions (50 in each complexity type). The system achieves an F-measure of 66.83% for English and 40.36% for Spanish.

Summary

The research in this area began with the concern of having annotated corpora as show by the work of Radev and Sundheim (2002), Schilder and Habel (2003) and Ahn *et al.* (2006). There are several approaches developed to answer temporal questions: Patterns, Templates, Algebra, First order logic prover, Decomposition, etc. We found no predominant approach, since each researcher chose a different path depending on the type of temporal questions.

Some works use the Web as a corpus to answer temporal questions such as Ahn *et al.* (2006); Paşca (2008); Tao *et al.* (2010) and Yahya *et al.* (2013b). However, these works try to extract information from the Internet and store it into knowledge bases to enable the QA system to access information more easily. The problem with this approach is that it needs constant effort to keep the database up to date.

Table 2.5 shows the related work on QA system focusing on Temporal Questions in a summarized form.

System	Corpus	Main Approach	Temporal Question Type
Paşca (2008)	Repository build from internet	Hand-build patterns	Simple
Tao <i>et al.</i> (2010)	OWL Ontology	API to access the ontology	Complex
Yahya <i>et al.</i> (2012)	knowledge base	SPARQL	Temporally restricted
Harabagiu and Bejan (2005)	Corpus annotated with TimeML	Template Question	Complex
Schockaert <i>et al.</i> (2006)	Data Collection	Algebra	Complex
Moldovan <i>et al.</i> (2005)	Logical Representation extract from texts	First order logic prover	Complex
Saquete <i>et al.</i> (2009)	TREC Corpus	Decomposition Question	Simple, Complex and Temporally restricted
<i>LX-ListQuestion</i>	<i>Web</i>	<i>Redundancy +Heuristics</i>	<i>Temporally restricted</i>

Table 2.5: Temporal QA summary

Our goal is to answer List questions with temporal restriction using the Web as corpus without any pre-processing and stored information. The information will be selected over plain-text. We did not find any system with the same characteristics as LX-ListQuestion.

2.8 Summary

This Chapter presented a review of the current state-of-the-art. First we presented a background over QA Competitions and the measures used to evaluate QA systems. We divided the presentation of the approaches in four topics: Portuguese QA systems, List QA systems, Web-based QA systems and Temporal QA systems.

We presented several QA systems developed for Portuguese. Some of these systems were developed to participate in competitions like CLEF, GikiCLEF or Páxico. The main approaches exploit (i) NLP tools and linguistic resources; (ii) the relation between question and possible answers; and (iii) the semantic content.

We discuss approaches of QA systems that focus on answering list questions. Statistical and machine learning approaches have been used to handle list questions. Other approaches that exploit Semantic Content from Wikipedia are also frequently used.

The Web-based QA systems are presented, which a great variety in the approaches developed that use Web as a corpus, such as those that explore redundancy, probabilistic models, clustering, etc. Most systems answer only factoid questions.

Finally, we presented the approaches to answer Temporal questions. Our overview indicates that a predominant approach did not emerge yet since each researcher chose a different path depending on the type of temporal questions. Some works use the Web as a corpus to answer temporal questions, however these works extract information and store it into knowledge bases to a QA system access the information easily.

3

Answering List Questions

The general consensus defines Open-Domain QA as aiming to provide the correct answer by exploring large collection of documents. The Web has become several orders of magnitude larger than any other document collection. As such, the availability of such a tremendous data resource has lead to the emergence of Web-based QA. The research on Web-based QA began 15 years ago and still growing. Recent Web-based QA systems are capable of handling factoid questions, but they lack the ability to produce answers to other type of questions.

This dissertation is a first step towards covering this gap, by presenting an approach to answer List questions using the Web as corpus. We developed our proposal to investigate appropriate ways to answer List questions processing and rendering information spread over multiple documents exploiting the redundancy of information available in the Web combining with heuristics aiming to improve the current state-of-the-art. Our proposal ensure that the system handles List questions of multiples types: (i) syntactically correct interrogative sentences; (ii) imperative sentences or (iii) keyword-based queries.

A major goal of this dissertation is to implement our approach to answer List questions as a fully-fledged Web-based QA system that provides a list of answers extracted from multiple documents. Build a system in such way that it does not require pre-indexing of documents, thus allowing it to provide answers in real time and make sure that the system can handle noisy and unstructured data, thus allowing it to run directly over the Web.

3. ANSWERING LIST QUESTIONS

3.1 Outline

This chapter is organized as follows. Section 3.2 provides an overview of List questions and their different features. The key challenges to this dissertation are remind in Section 3.3.

Our overall approach to answer List questions is discussed in Section 3.4. We present the design features of LX-ListQuestion, the questions that are expected as input and the supporting tools used to build our system.

In Section 3.5 we describe in detail the architecture of the system which all modules are described and examples are discussed. Finally in Section 3.6, we present the online architecture and user interface of our implementation of Open-domain Web-based QA system: LX-ListQuestion.

3.2 List Questions

The main difference between List questions and Factoid questions is that for Factoid questions there is only one correct answer, while List questions require a list of correct answers, making the task of correctly answering them more complex. We performed a corpus study that allowed us to identify three basic forms of List questions: (i) **Interrogative Questions**: a question starting with an interrogative pronoun (Where, Which, Who, etc.); (ii) **Imperative Questions**: a question starting with an imperative verb (List, Name, Say, etc.); (iii) **Others**: a question without an interrogative pronoun or an imperative verb, usually in the form of complex noun phrase. Table 3.1 shows some examples of these various forms of List questions.

Type of List Question	Examples
Interrogative Questions	Which countries adopted the Euro? Which presidents were born in 1945? Who are the founders of Intel?
Imperative Questions	Name all Apollo 14 astronauts. List all companies in Munich. List the children of Margaret Thatcher.
Others	Communist countries. Soccer clubs in Spain. All movies with Tom Cruise.

Table 3.1: Examples of list questions

3.2 List Questions

Our corpus study also indicates that we can categorize List questions according to their complexity. It is usually easier to answer short questions, while longer questions with increased linguistic sophistication are more complex to process and correctly answer. Table 3.2 shows some examples of List questions with different levels of complexity.

Simple/Complex Questions	Examples
Simple	Give me all movies with Tom Cruise.
Simple	Give me all films produced by Hal Roach.
Simple	Give me all books written by Danielle Steel.
Complex	In which films did Julia Roberts as well as Richard Gere play?
Complex	List all episodes of the first season of the HBO television series The Sopranos.
Complex	Which companies work in the aerospace industry as well as in medicine?

Table 3.2: Examples of simple and complex list questions

List questions can be made even more complex by adding constraints. The most common constraints are:

- Temporal constraints, related to time (months, years, centuries, etc.);
- Geographic constraints, related to localization (cities, region, countries, continents and so on);
- Quantitative constraints, related to numerical quantities such as amounts and frequencies.

Table 3.3 shows some examples of List questions with constraints.

Constraints	Examples
Temporal	Which presidents were born in 1945? Which organizations were founded in 1930? Give me all libraries established earlier than 1400.
Geographic	Give me all cars that are produced in Germany. List all companies in Munich. Give me a list of all lakes in Denmark.
Quantitative	Which caves have more than 3 entrances? Which German cities have more than 250000 inhabitants? Give me the websites of companies with more than 500000 employees.

Table 3.3: Examples of list questions with constraints.

3. ANSWERING LIST QUESTIONS

Identifying the question focus is an important task in Question Answering. Our study shows that the focus can be expressed by a named entity or common noun. A question may have a single or multiple focus. Table 3.4 shows some examples of List questions with Question-focus with different features.

Question-Focus		Example
Part-of-speech	Type	
Common Noun	Single	Which are the ingredients of gun metal?
Named Entity	Single	Which states border Illinois?
Named Entity	Single	Give me all federal chancellors of Germany.
Named Entity	Multiple	Give me all airports, bridge and highways located in California, USA.
Named Entity	Multiple	List all newspapers, magazines and books published on New York.

Table 3.4: Examples of list questions with question-focus.

Turning now the focus on the answers to List questions, our study indicates that they may appear in many places and in many forms: the answers can be in the same sentence when the sentence is already an enumeration of the answers or the answers can be spread over multiple sentences or even multiple documents. In the latter case, a QA system that handles List questions needs to find all the answers spread over the several texts and compose the final list of answers. Table 3.5 shows some examples of List questions which the answers are found in the same sentence and spread over several documents.

Example of Answers to List questions in the same sentence:

Question: Give me all members of Prodigy.

Sentence: The current members include **Liam Howlett** (composer), **Keith Flint** (vocalist), **Leo Crabtree** (drums), **Rob Holliday** (guitarist) and **Maxim Reality** (vocalist).

Example of Answers to List questions in the different documents:

Question: Give me all movies with Tom Cruise.

Document 1: Cruise played a fighter pilot in action drama **Top Gun** and also starred opposite Paul Newman in the drama **The Color of Money**.

Document 2: In 1988, Tom Cruise played with Dustin Hoffman winning drama **Rain Man** and also, in the same year, the winning romantic drama **Cocktail**.

Document 3: **Born on the Fourth of July** (1989), Tom Cruise as anti-war activist Ron Kovic in the drama adaptation of Kovic's memoir.

Document 4: In 1999, Cruise starred in the Stanley Kubrick-directed erotic thriller **Eyes Wide Shut** opposite his then wife Nicole Kidman, and also appeared in the drama **Magnolia**.

Table 3.5: Examples of answering list questions

3.3 Challenges

The challenges have already been thoroughly discussed in Section 1.5. Here we remind the key challenges involved in the implementation of an Open-domain Web-based QA system that handles List questions:

- **Quality:** The process of answering questions using the Web as a corpus is complicated by the low average quality of documents. Due to how easy it is to publish something on the internet, many documents are poorly written or simply contain incorrect information. As a result, an answer extracted from these Web documents cannot be trusted as the correct answer.

This issue can be alleviated through data redundancy, which is the main approach of our dissertation, since multiple occurrences of the same answer in different documents lends credibility to that answer.

- **Timeliness:** Using the Web as a corpus instead of a database allows giving the user an answer even when the question refers to recent events or facts which otherwise would require constant effort to maintain an up to date database.

Providing an answer to a question in real-time when running over the Web is a key challenge in the context of this dissertation. This is a challenge covered in this dissertation.

- **Accuracy:** The answer precision of a QA system is important an aspect. Some researchers believe that incorrect answers are worse than no answers and producing accurate answer to List questions is a big challenge in this dissertation.
- **Completeness:** Getting a complete list of answers is desirable. In most cases the answers are spread over several documents and composing a complete list of answers is challenging.

When the list of answer is composed by two or three items is more easy to find the complete list of answers, although when the answer list is composed by 50 or 100 items the difficult of this task increase.

3. ANSWERING LIST QUESTIONS

3.4 Approach

Our approach allows to collect answers from multiple documents to compose the final list of answers. The key feature of our approach is that it exploits the redundancy of information available online combined with heuristics to improve QA accuracy.

Driving Insight

Approaches to Information Retrieval, and to related tasks such as Question Answering, that use only one frequency threshold face the common precision-recall tradeoff problem: if the threshold is high the precision increases and recall decreases; if the threshold is low the recall increases and the precision decreases. To find a perfect single threshold that gives the best balance between recall and precision is not an easy task. This suggests that we need a more informed threshold that in some way links frequency with relevance between the sentence and question.

This can be done in several ways: We opted for a simple elegant solution which separates the candidates into two lists based on relevance between the sentence and question. Each list will be filtered by a different threshold. Our strategy applies two separate thresholds: one more relaxed for high relevance candidates and another more stringent for low relevance candidates. This strategy that separates the candidates in two lists and applies different thresholds leads to a better balance between precision and recall.

Below we present more details of our approach:

1. **Bipartite List Approach:** We build two lists: (i) Premium List, which is composed by candidates extracted from the sentences with high relevance to the question and (ii) Work List, which is composed by candidates extracted from the sentences with weak relevance to the question. Our approach exploits redundancy to find all answers to the List questions, and uses their frequency as a factor to select the correct answer.

In each list, the candidates that appear repeated are grouped together and their frequency is calculated. Following this, two frequency thresholds are calculated. One threshold is used to filter the candidates of Premium List and the other to filter the candidates of Work List. The threshold applied to the Premium List is relaxed since these candidates are more credible on account of them having been extracted from

sentences classified with high relevance score. The threshold applied to the Work List is stringent since these candidates were extracted from noisy sentences and only the candidates with high frequency go through to the final list of answer.

2. **Heuristics:** We developed and applied three heuristics based on word occurrence.

- Verb-Rule, which selects a candidate as an answer if the candidate appears in a sentence with the same verb given by the question.
- Title-Rule, which selects as an answer all candidates from documents whose title matches the question.
- Sentence Match-Rule, which selects as answer all candidates extracted from sentences that match the question.

Combining bipartite list and heuristics is a novel approach that, to the best of our knowledge, has not been attempted before.

Design Features

A major goal of this dissertation is to implement our approach to answer list questions as a fully-fledged Web-based QA system that provides a list of answers extracted from multiple documents. Our implementation is guided by the following design features:

- Exploits redundancy to find answers to List questions;
- Compiles and extracts the answers from multiple documents;
- Collects at run-time the documents from Web using a search engine;
- Provides answers in real time without resorting to previously stored information.

3.4.1 Expected Input

Our system expects as input questions in Portuguese that require a list of answers and accepts all three basic forms of List questions: (i) a question starting with an interrogative pronoun; (ii) a question starting with an imperative verb and (iii) a question that starts in the form of complex noun phrase.

3. ANSWERING LIST QUESTIONS

The system handles questions with different level of complexity: (i) simple and usually short questions (no longer than three keywords) and (ii) complex question with some linguistic sophistication. LX-ListQuestion answers questions which expect named entities as an answer.

Table 3.6 shows examples of List questions which LX-ListQuestion is capable of finding the answer to:

Question Type	Simple/Complex	Examples
Interrogative Questions	Simple	Quais são as ilhas em Moçambique? <i>Which are the Mozambican islands?</i>
	Complex	Quem são os escritores cabo-verdianos com obra publicada em crioulo? <i>Who are the Cape Verdean writers with published work in creole?</i>
	Complex	Em que cidades portuguesas têm festivais medievais? <i>Which Portuguese cities have medieval festivals?</i>
Imperative Questions	Simple	Liste os parques nacionais em Moçambique. <i>List the National Parks in Mozambique.</i>
	Complex	Nomeie os escritores moçambicanos que receberam o Prémio Camões. <i>Name the Mozambican writers that received the Camões Prize.</i>
	Complex	Mencione as cidades que fizeram parte do domínio português na Índia. <i>Name cities that were part of the Portuguese Empire in India.</i>
Other	Simple	Capitais das províncias de Angola <i>The capitals of Angolan provinces.</i>
	Complex	Cidades lusófonas conhecidas pelo seu carnaval. <i>Lusophone cities known for their carnival celebrations.</i>
	Complex	Países que venceram a Copa do Mundo em uma disputa de pênaltis. <i>Countries that won the World Cup by penalty shootouts.</i>

Table 3.6: Examples of list questions expected input of LX-ListQuestion.

3.4.2 External Resources and Support Tools

Our system was built with the support of some tools. In this section we describe the tools used.

1. LX-Conjugator¹: is a tool for conjugation of Portuguese verbs (Costa, 2004). The system takes an infinitive verb form and delivers the corresponding conjugated forms. The Portuguese verbal inflection is a most complex part of the Portuguese morphology given the high number of conjugated forms for each verb (ca. 70 forms in non pronominal conjugation).

¹<http://www.lxcenter.di.fc.ul.pt/services/en/LXServicesConjugator.html>

2. LX-Suite¹: is a system for shallow processing of Portuguese (Branco and Silva, 2006). The system is based on a pipeline of several tools. The tools for lemmatization and morphological analysis are inserted at the end of the pipeline and are fed by three other tools: a sentence splitter, a tokenizer and POS tagger.
3. LX-Ner²: is a tool for recognition of expressions for named entities (Ferreira *et al.*, 2007) in Portuguese. The name entities are classified in Persons (PER), Organization (ORG), Location (LOC), Events (EVT) and works (WRK).
4. Multi-WordNet PT (MWNPT)³: is a lexical semantic network for Portuguese language. The database was shaped under the ontological model of wordnets. It spans over 17,200 concept/synsets, linked under the semantic relations of hyponymy and hipernymy. These concepts are made of over 21,000 word senses/word forms and 16,000 lemmas. It includes the subontologies under the concepts of Person, Organization, Event, Location and Work Arts.
5. TEP⁴: Electronic Thesaurus for Brazilian Portuguese (Maziero *et al.*, 2008). The TEP database stores sets of synonym and antonym for the word forms. We use the database to improve keywords expansion.
6. Google Custom Search⁵: The Google Custom Search is an application programming interface (API) that allow retrieve and display search results from Google Custom Search. The API works integrated into the system application. The API provides 100 search queries per day for free.
7. HTML Parser⁶: HTML Parser is a Java library used to parse HTML. The library allow to transform HTML pages into plain text and also to create and edit pages. The library can be coupled with the QA system being development.

¹<http://www.lxcenter.di.fc.ul.pt/services/en/LXServicesSuite.html>

²<http://lxcenter.di.fc.ul.pt/services/en/LXServicesNer.html>

³<http://lxcenter.di.fc.ul.pt/services/en/LXServicesWordnet.html>

⁴<http://www.nilc.icmc.usp.br/tep2/>

⁵<http://www.google.com.br/cse/>

⁶<http://htmlparser.sourceforge.net/>

3. ANSWERING LIST QUESTIONS

3.5 Architecture

Our architecture follows the basic architecture of QA system proposed by Paşca (2003) had been composed by three main modules: Question Processing, Passage Retrieval and Answer Extraction, as showed in Figure 3.1.

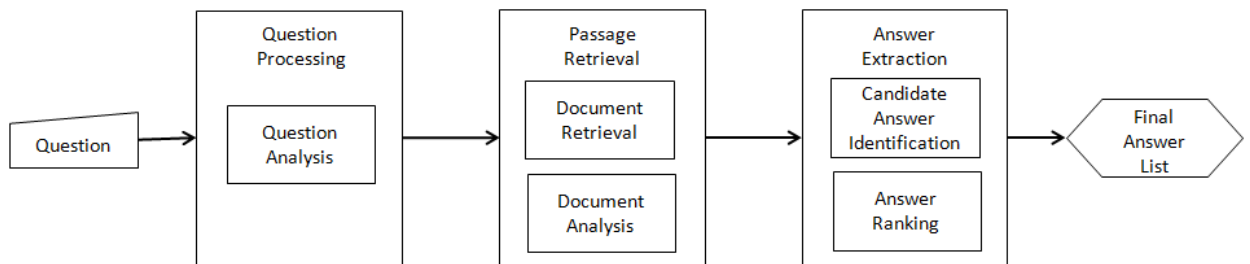


Figure 3.1: Question answering system architecture

This is a formal QA system architecture widely accepted and provides a simple one-way dataflow of the processing. This modularity engage particularity implementation hiding a high complexity when considering a development of a Open-domain Web-based QA system. Each module is deeply discuss in this Section.

3.5.1 Question Processing

The Question Processing module is responsible for converting a natural language question into a format that a computer is capable of further handling. Figure 3.2 shows the main components of the Question Processing module.

The process of question processing relies on the following sub-tasks: (1) question analysis; (2) extraction of keywords; (3) transformation of the question into a query for the search engine; (4) identification of the semantic category of the expected answer; (5) counting of the number of words in the question; and (6) identification of the question-focus.

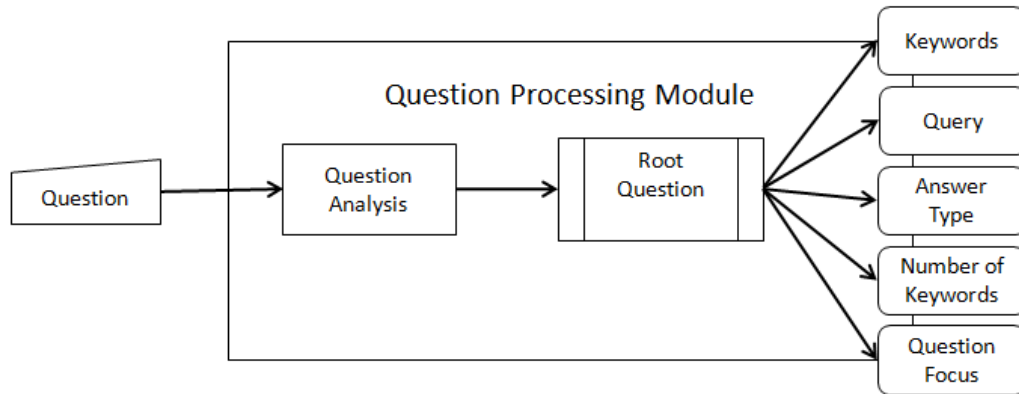


Figure 3.2: Question processing module

3.5.1.1 Question Analysis

The Question Analysis sub-task is responsible for identifying patterns occurring in questions and for clearing questions leaving only the significant part called root-question. In the context of QA research, list questions may appear in three basic forms (see Table 3.7 for examples):

1. Sentence without an interrogative pronoun;
2. Question that begins with an interrogative pronoun;
3. Sentence that begins with an imperative verb form.

Question Type	Example
Without pronoun	Províncias toscanas que produzem Chianti.
Interrogative pronoun	Quais as províncias toscanas que produzem Chianti?
Imperative verb	Liste as províncias toscanas que produzem Chianti.

Table 3.7: Examples of different type of list questions.

3. ANSWERING LIST QUESTIONS

When the pattern is identified, the interrogative pronoun or the imperative verb are discarded. What remain we named *root question* and will be processed in the next step. Some examples follow below:

Sentence Examples:

Quais as províncias toscanas que produzem Chianti?
Liste as províncias toscanas que produzem Chianti.

All sentences above will be reduced to this root question:

Províncias toscanas que produzem Chianti.

3.5.1.2 Transformation of the Question into a Query

In this processing step, the root question will be transformed into a query for the search engine. The root question will be processed by LX-Suite which, among other linguistic information will assign a part-of-speech and lemma to each word, as exemplified below:

Root question:

Províncias toscanas que produzem Chianti.

Words of root question and lemmas:

províncias/PROVÍNCIA toscanas/TOSCANA que/QUE produzem/PRODUZIR Chianti/CHIANTI

The query will be composed by the lemmas of the root question, but only some part-of-speech categories will be retained, namely proper names, common nouns, verbs and adjectives. Other categories, such as articles, prepositions and pronouns are discarded.

Complete query:

PROVÍNCIA + TOSCANA + PRODUZIR + Chianti

3.5.1.3 Keywords Extraction

The keywords will be used to select relevant passages from source texts. To gather the keywords, we use the technique of word expansion by resorting to the lexical databases MWNPT & TEP (to expand common nouns) and LX-Conjugator tool (to expand verbs). The word expansion only applies to common nouns and verbs.

Nominal Expansion: The algorithm will identify common nouns in the root question and their synonyms will be retrieved from the lexical databases MWNPT and TEP.

Following the example:

Root question:

Províncias toscanas que produzem Chianti

For instance, the word “província” is a common noun and when this word is searched in the lexical database its synonyms are retrieved:

Synonyms of “província”:

região, zona, distrito, lugar

Verbal Expansion: The algorithm identify verbs in the root question and the LX-Conjugator tool will take an infinitive verb form and deliver it conjugated in the Past Perfect and Past Participle forms, which are the forms that most commonly appear in List questions.

An example follows:

conjugated forms of “produzir”:

produzem, produziu, produzido

Building the full set of keywords: The keywords will be composed by the (1) root question, (2) lemmas, (3) the synonyms of the common nouns in the root question and (4) the conjugated forms of the verbs in the question:

Full set of keywords:

províncias, província, toscanas, toscana, produzem, produzir, produziu, produzido
Chianti, região, zona, distrito, lugar

3.5.1.4 Semantic Category of the Expected Answer

The semantic category of the question is determined using MWNPT and will be used by the Answer Extraction Module (see Section 3.5.3). This information is important to check if the candidates has the expected semantic category of the question.

Following the example in Table 3.8, the semantic category of the word “província” is “LOC” (meaning LOCATION). In this case, the list of answers will be composed by locations. The other possible semantic categories are: PER (person), ORG (organization), EVT (event) and WRK (work). If a given word is not present in MWNPT, the system assumes the value UNK (unknown) and all categories of named entities are considered for processing.

3. ANSWERING LIST QUESTIONS

Word	MWNPT	
	Synonyms	Semantic Category
província	região, zona, distrito, lugar	LOC

Table 3.8: Semantic category of the expected answer

3.5.1.5 Counting Keywords

The number of keywords is used to calculate the threshold for classifying sentences according to their relevance to the question. We also apply heuristics that recognize named entities on the basis of the part-of-speech tags that were assigned by LX-Suite. It is worth mentioning that the named entities are counted only once and articles and prepositions are discarded from the counting. Table 3.9 shows examples of the counting of keywords.

Number of Keywords	Questions
3	{Filmes} {brasileiros} sobre {futebol.}
4	{Províncias} {Toscana} que {produzem} {Chianti}
5	Liste os {eventos} onde {Maria de Lurdes Mutola} {foi} {medalha} de {ouro}

Table 3.9: Examples of counting keywords

3.5.1.6 Identifying the Question-Focus

Moldovan *et al.* (2000) defines the question-focus as been a word or a sequence of words which define the question and disambiguate the question by indicating what the question is looking for. In the context on this work, we developed a simple heuristic for determining the question-focus from the question. We consider the question-focus to be the first common noun of the question. Table 3.10 shows some examples.

Questions	Question-Focus
Filmes brasileiros sobre futebol.	Filme
Províncias Toscana que produzem Chianti	Província
Liste os eventos onde Maria de Lurdes Mutola foi medalha de ouro.	Evento

Table 3.10: Examples of question-focus identification

3.5.2 Passage Retrieval

The Passage Retrieval module is responsible for searching web pages based on the query retrieved and saving the Web-pages into local files for pre-processing. This module is also responsible for separating the content from HTML tags and selecting the relevant information related with the input question. It has two main tasks: document retrieval and document analysis. Figure 3.3 shows the Passage Retrieval Module and its components.

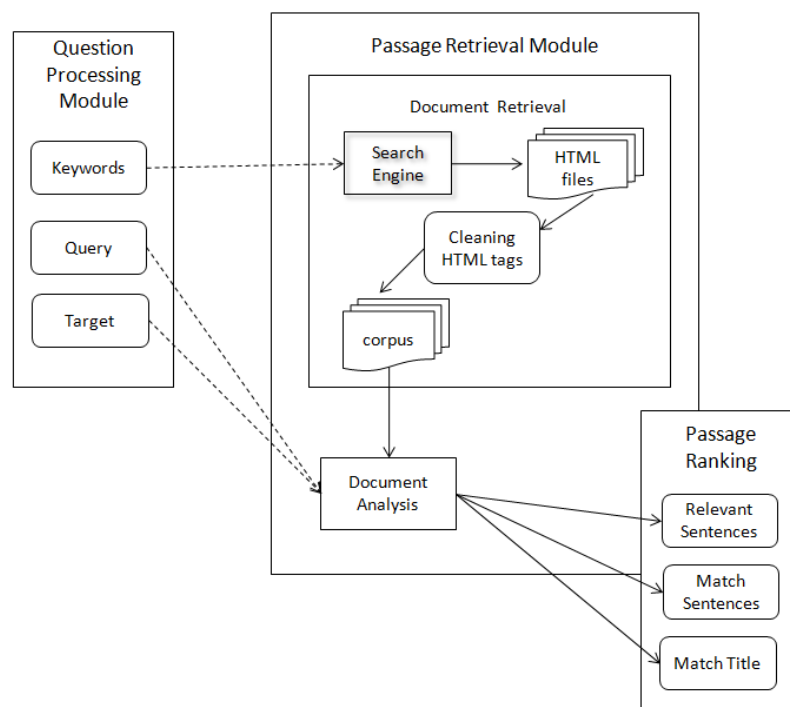


Figure 3.3: Passage retrieval module

3.5.2.1 Document Retrieval

In this task the documents related with the questions are retrieved from the Web. We only use HTML files from Web. Other types of files are discarded (e.g. .doc, .pdf, .ppt).

The files are downloaded and stored on the server for textual processing. The documents are retrieved using the query generated by the Question Processing Module (Section 3.5.1), according to the steps detailed below:

3. ANSWERING LIST QUESTIONS

1. send query: submitting the query to the Google Search Engine¹;
2. get links: the search engine returns links from the top 10 documents;
3. download files: verify if the link is a HTML file. If so, download and save the document;
4. clean files: separate HTML markup from content using HTML Parser.

3.5.2.2 Document Analysis

In this task the documents retrieved by the previous task will be analyzed and the relevant information will be selected. Initially the text is processed by a Segmenter tool, which will markup the text and split it into sentences. Figure 3.4 shows an example.

Before Sentence Segmentation
O Classico no centro de Chianti, através da províncias de Florença e Siena. Arentino Colli na província de Arezzo. Colli Senesi sul de Chianti Classico, nas colinas de Siena, esta é a maior das sub-regiões. Colline Pisane sub-zona oeste, na província de Pisa. Montespertoli localizado dentro do Colli Fiorentini. Inúmeras vezes a vi emoldurando restaurantes italianos, principalmente em São Paulo. Já a qualidade deste vinho variou em muito.
After Sentence Segmentation
O Classico no centro de Chianti, através da províncias de Florença e Siena. Arentino Colli na província de Arezzo. Colli Senesi sul de Chianti Classico, nas colinas de Siena, esta é a maior das sub-regiões. Colline Pisane sub-zona oeste, na província de Pisa. Montespertoli localizado dentro do Colli Fiorentini. Inúmeras vezes a vi emoldurando restaurantes italianos,principalmente em São Paulo. Já a qualidade deste vinho variou em muito.

Figure 3.4: Before and after the segmenter tool.

¹The Google Search Engine is configured to return only pages written in Portuguese.

After the Segmenter tool is applied, three important steps are performed (Figure 3.5 offers a diagram of the process):

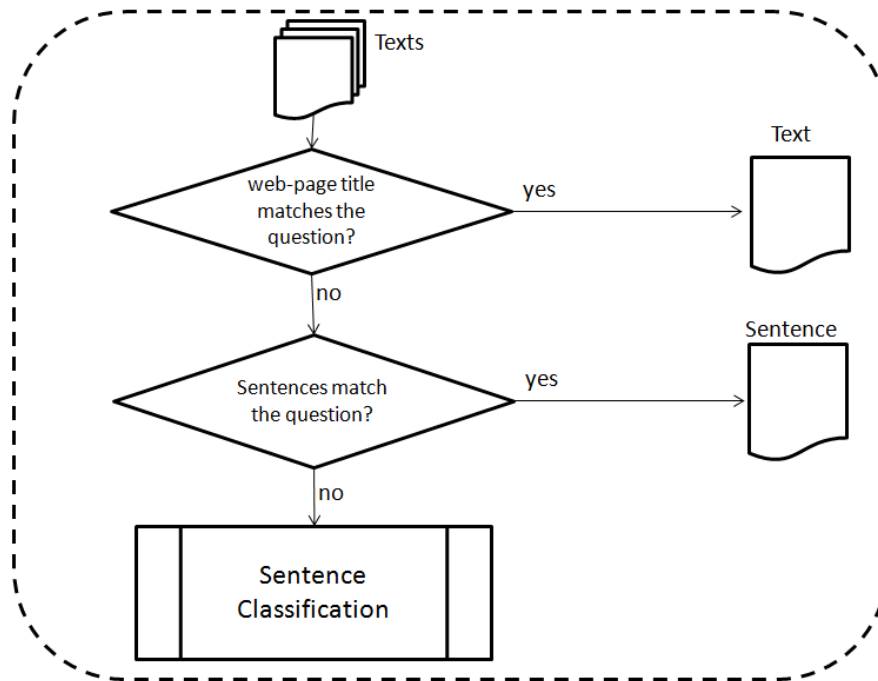


Figure 3.5: Document analysis process

1. Verify title of text: If the HTML title of the text matches the root-question (that is, it contains the same words in the same order), the text is set apart to be sent to the next module. Example:

Root question: “Províncias Toscanas produzem Chianti.”.

Web-page title: “As famosas **províncias Toscanas** que **produzem** o melhor do vinho **Chianti**”.

In Figure 3.6, the words in the title, “províncias”, “Toscanas” “produzem” and “Chianti.”, appear in the same order as in the root-question.

3. ANSWERING LIST QUESTIONS



Figure 3.6: Web page title matches the question.

2. Verify if there are some sentences that match the root-question: If a sentence matches the root-question, the sentence is set apart to be processed by the next module.

Example: Root question: "Províncias Toscanas produzem Chianti"

Sentence: "As **províncias Toscanas** que atualmente **produzem** o vinho **Chianti** são: Florença, Pisa e Pistoia."

In the Figure 3.7, the words in the highlighted sentence, namely "províncias", "Toscanas" "produzem" "Chianti" appears in the same order as in the root-question.

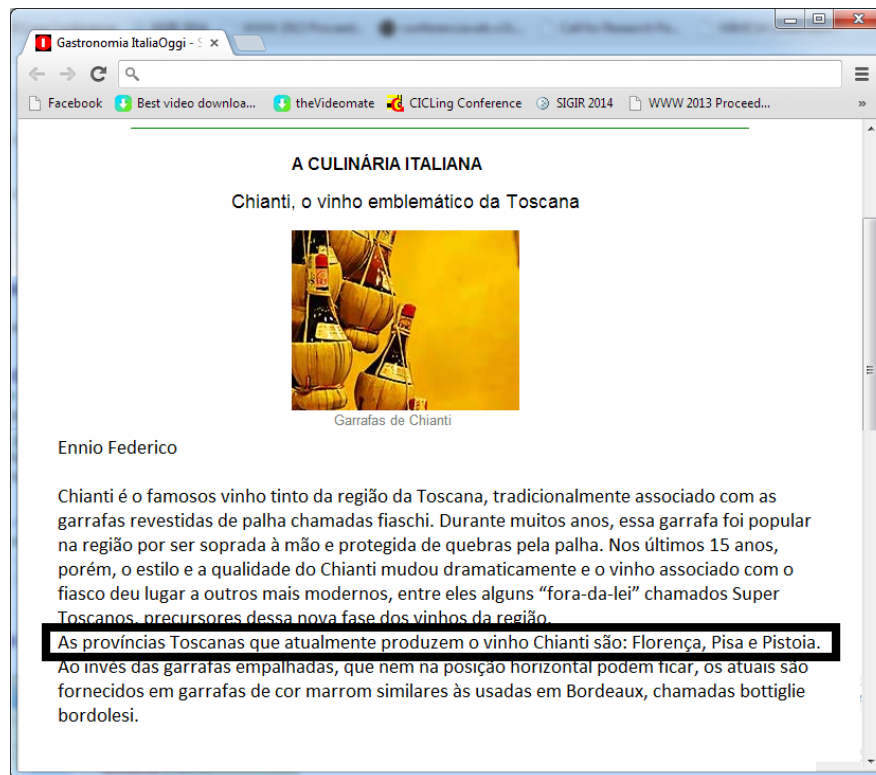


Figure 3.7: Sentence matches the question.

3. Sentence classification: The sentences are classified in three classes according to their relevance with respect to the root question, which depends on a Relevance Threshold, set at half the number of keywords. Depending on their classification, the sentences are stored in distinct sets. A sentence will have “weak” relevance if it contains less keywords than the Relevance Threshold; “medium” relevance if it contain as many keywords as the Relevance Threshold; and “strong” relevance if it contains more keywords than the Relevance Threshold.

For instance, the root question “Províncias Toscanas que produzem Chianti.” has 4 keywords, giving a Relevance Threshold of 2 (half of the number of keywords). Table 3.11 shows examples of sentences classified based on this Relevance Threshold.

3. ANSWERING LIST QUESTIONS

Sentence	Keywords	Class
O grande problema é que somente a palavra Chianti diz pouca coisa sobre o vinho.	Chianti	weak
Chianti Rufina na parte nordeste da região situada em torno do município de Rufina.	Chianti, região	medium
O vinho Chianti é produzido em uma região muito vasta da Toscana que compreende as províncias de Firenze, Siena, Prato, Arezzo e Pistoia.	Chianti, Toscana, região, província	strong

Table 3.11: Sentence classification.

3.5.3 Answer Extraction

The Answer Extraction Module aims at identifying and extracting relevant answers from the sentences previously classified and presenting them in the form of a list. This module uses information provided by previous modules. The Expected Answer Type identified during question processing will guide the identification of candidate answers. Similarly, the sentences previously classified help limit the search space for candidate answers. Figure 3.8 presents a diagram of the Answer Extraction Module.

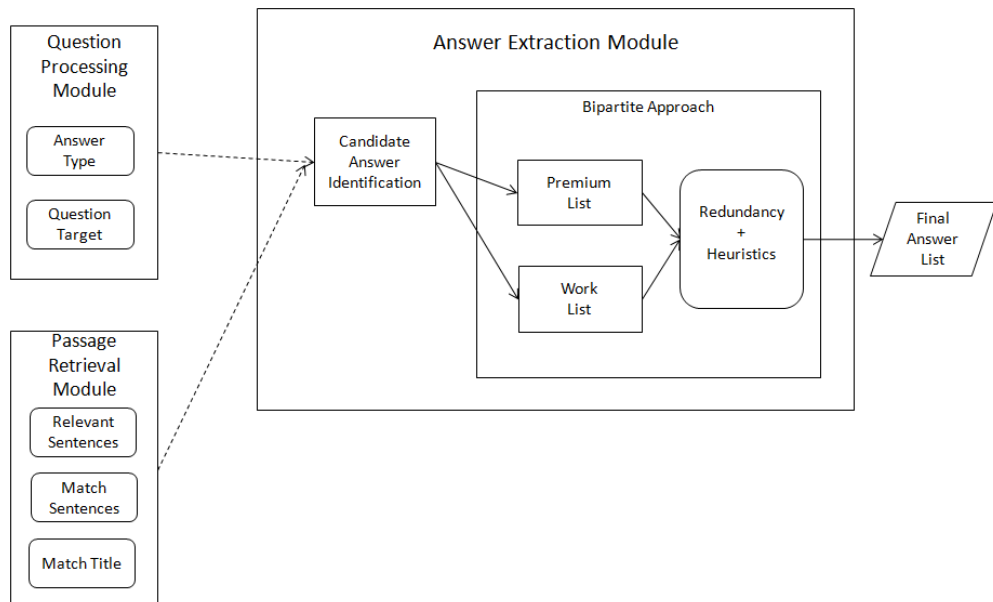


Figure 3.8: Answer extraction module.

3.5.3.1 Extracting Candidate Answers

We developed two approaches to extract candidate answers: (i) an approach based on a Named Entity Recognizer and (ii) an approach based on the Question-Focus:

Approach based on Named Entity Recognizer: The candidate answers are a subset of the words that are extracted from the passages and identified by the Named Entity Recognizer (NER), named LX-Ner. This approach is common in QA and usually provides the best precision since it ensures that the semantic category of the candidate matches that of the expected answer. However, this ties the performance of answer extraction procedure to the performance of the underlying NER tool. A classification error during the NER phase will impede the successful retrieval of candidates. Figure 3.9 shows candidates extracted from the sentence:

“Chianti Colli Senesi é produzido numa subzona situada ao sul da região do Chianti Clássico, nas colinas da Província de Siena.”

The candidates are extracted using a hand-pattern build based on tag:

<NAMEX TYPE="LOC">(meaning LOCATION) given by LX-Ner.

<p style="text-align: center;">Sentence processed by LX-Ner:</p> <p><NAMEX TYPE="PER">Chianti/PNM Colli/PNM Senesi/PNM</NAMEX>É/V produzido/PPA em/PREP a/DA subzona/CN situada/PPA a/PREP o/DA sul/CN de/PREP a/DA <NAMEX TYPE="LOC">região/CN de/PREP o/DA Chianti/PNM Classico/PNM</NAMEX> ./PNT em/PREP as/DA colinas/CN de/PREP a/DA <NAMEX TYPE="LOC">província/CN de/PREP Siena/PNM</NAMEX></p>
<p style="text-align: center;">Candidates extracted using the hand-build pattern: <NAMEX TYPE="LOC"></p> <p><NAMEX TYPE="LOC">região/REGIÃO/CN#fs e_/PREP o/DA#ms Chianti/PNM Classico/PNM </NAMEX> <NAMEX TYPE="LOC">província/PROVÍNCIA/CN#fs de/PREP Siena/PNM </NAMEX></p>
<p style="text-align: center;">Candidates without the tags:</p> <p>região do Chianti Clássico província de Siena</p>

Figure 3.9: Example: candidates extracted by LX-Ner

Approach based on the Question-Focus: In this approach, the candidate answers is a subset of the words that are extracted from the passages that matches question-focus based on hand-build pattern. The pattern will select as a candidate all words that appear after the question-focus. Figure 3.10 shows candidates extracted based on question-focus: “Província”.

3. ANSWERING LIST QUESTIONS

Sentences:
Chianti Colli Aretini Produzido em uma área da província de Arezzo, é um vinho jovem fresco e levemente frisante, ideal para ser consumido ainda jovem. A comuna italiana da região da Toscana, província de Siena, parece ter saído de um filme da Idade Média. Suas videiras se concentram principalmente na província do Piemonte, na Itália, numa zona onde as uvas Moscato dão o melhor de si.
Candidates extracted by Question-Focus: “Província”
província de Arezzo província de Siena província do Piemonte

Figure 3.10: Example: candidates extracted by question-focus

3.5.3.2 Building the Answer List

We introduced our approach in Section 3.4, in this Section we explain in detail our implementation of the approach to answer List questions. The process of building the List Answers is based on a redundancy and the strategy here is to take advantage of the sentences previously classified by the Passage Retrieval Module.

How we explained earlier, our bipartite approach is based on building two lists and applying two thresholds, one for each list. The main elements that will compose the Premium List are taken from sentences previously classified as highly relevant and will serve to guide the rest of the processing. If we were to consider only these elements, the list of answers would probably contain correct items.

However, the list may be incomplete and lack elements. Then, continuing in the same vein of our strategy, we build the Work List using the candidates extracted from the sentences classified as medium and low relevance. The Building the List Answer process based on Bipartite approach is detailed below:

1. The Premium List is built from candidates extracted from the sentences previously classified as highly relevant (see Section 3.5.2).
2. The Work List is built from candidates extracted from the sentences previously classified with medium or low relevance.
3. If the Premium List is empty, the candidates extracted from sentences classified as medium relevance will be used to build the Premium List, leaving only the candidates extracted from low relevance sentences to build the Work List.

4. The elements in each list that appear repeated are grouped together and their frequency is calculated.
5. Two frequency thresholds are calculated both from the Work List. One threshold will be used to filter the Premium List (t_P) and the other to filter the Work List (t_W). The thresholds are calculated by the following procedure:
 - Let, $wa = \frac{\sum_j c_j \times j}{\sum_j c_j}$ be the weighted average of the elements in the Work List, where j is the frequency and c_j is the frequency of elements with frequency j .
 - Let u be an (empirically determined) upper bound on the admissible values for j , in order to limit the impact of the elements with very low frequencies.
 - Let $\hat{c}_{u,j} = \min(j, u)$ be the frequency of frequencies bounded by u .
 - Let $\hat{wa} = \frac{\sum_j \hat{c}_{u,j} \times j}{\sum_j \hat{c}_{u,j}}$ be the weighted average of the elements in the Work List, taking into account frequency of frequencies bounded by u .
 - To calculate the threshold $t_P = \hat{wa}$ for the Premium List, u is set to 7 after experimentation.
 - To calculate the threshold $t_W = \hat{wa}$ for the Work List, u is set to 1 after experimentation.
6. Filter with t_P : Candidates in the Premium List are filtered using t_P . The frequency of each candidate is compared to the threshold t_P : for a candidate to pass to the Final List of Answers, its frequency should be equal to or greater than the threshold t_P .
7. Filter with t_W : Candidates in the Work List with a frequency above t_W threshold calculated are also included in the Final List of Answers.

Our approach combines redundancy available in the Web with Heuristics. The Heuristics implemented in our system is described as follow:

1. **Heuristic based on Question-Verb:** Candidates in the Premium List who were not promoted to the Final List of Answers, after applying Threshold t_P , still have a second chance to be included following the criterion of analysis of the Verb. If the sentence in which the candidate occurred contains the same verb of the question, then the candidate is included in the Final List of Answers.

3. ANSWERING LIST QUESTIONS

2. **Heuristic based on Document-Title:** All candidates extracted from the documents in which the title matches (i.e. all keywords are present) the root question pass to Final List Answers. The process to extracted these candidates was explained in Section 3.5.2.2
3. **Heuristic based on Sentence-Question-Match:** All candidates extracted from sentences that match the root question pass to Final List Answers. The process to extracted these candidates was explained in Section 3.5.2.2

Figure 3.11 summarizes the building of the list of answers process combining the Bipartite approach and Heuristics.

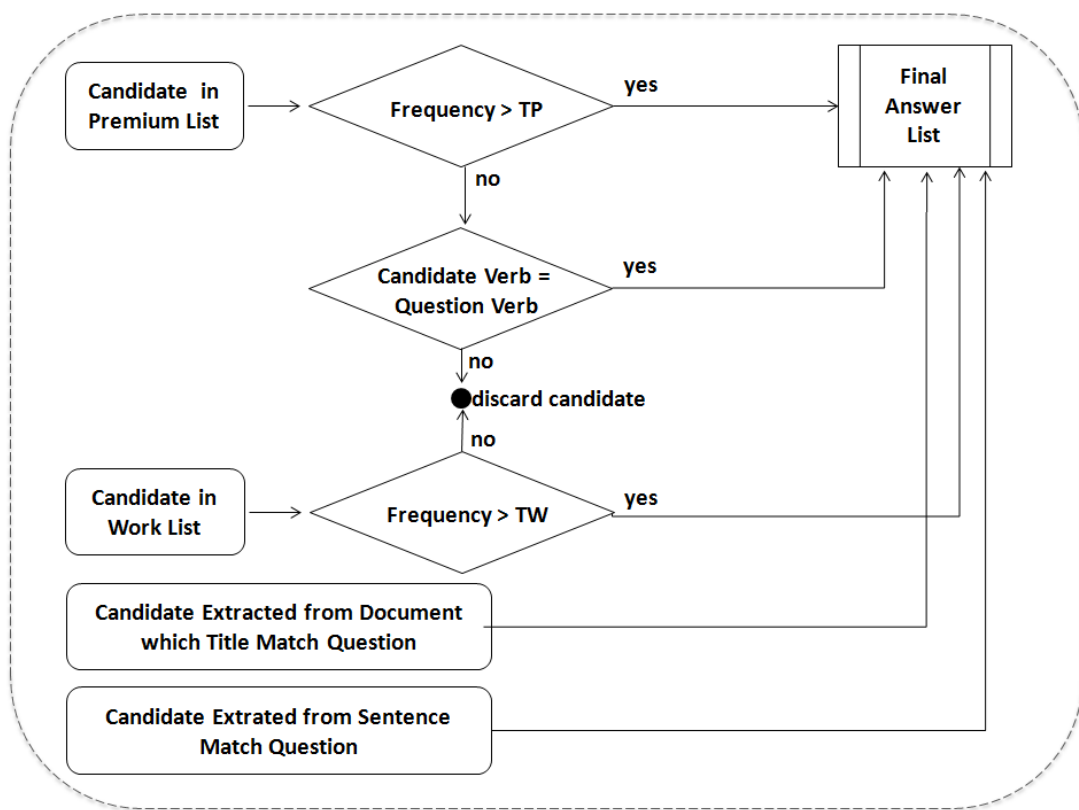


Figure 3.11: Building the list of answers.

3.6 LX-ListQuestion: an Open-Domain Web-based QA system for List Questions

To satisfy a goal of this dissertation, namely to provide an online Open-domain Web-based Question Answering system for List Questions, the LX-ListQuestion is available online at <http://lxlistquestion.di.fc.ul.pt>. In this Section we present the architecture underlying the online version of LX-ListQuestion.

Online Architecture

The online version of the LX-ListQuestion was developed combining Java Server Pages (JSP)¹ and Java². The LX-ListQuestion System is developed using Java and the user interface using the Java Server Pages that allows accessing Java programming language. Figure 3.12 shows a diagram of this architecture.

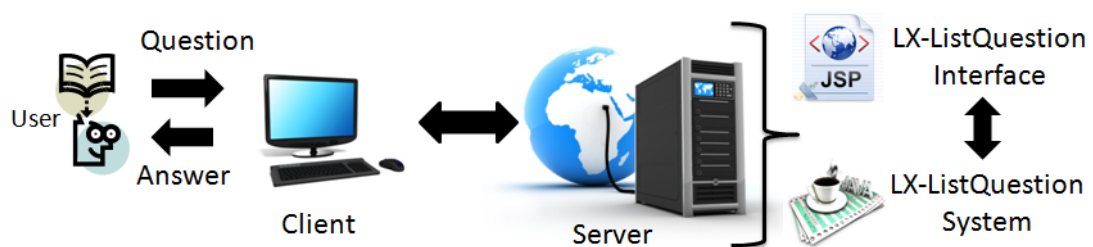


Figure 3.12: LX-ListQuestion - online version architecture.

User Interface

The user interface was developed using JSP because it allows an easy connection with the system in Java.

We sought to develop an interface easy to use, where there is a field that allows the user to write the question.

Before running the system the user can choose the following options:

¹<http://www.oracle.com/technetwork/java/javaee/jsp/index.html>

²<http://www.java.com/>

3. ANSWERING LIST QUESTIONS



Figure 3.13: LX-ListQuestion - online version interface - list answers.

- **Normal** or **Expandida**: The user can choose the *Normal* option for the normal processing system. If the user chooses the option *Expandida*, the system ignores the filters and the process returns all the candidates found in documents and their respective frequency.
- **Lista** or **WordCloud**: The system provides two ways of presenting the answers. The first is a traditional way where the answers are displayed in a numbered list. The answers at the top of the list are more relevant and the answers at the bottom of the list are less relevant. We also use the font size to distinguish the answers. The answers with the larger font size are the most relevant.

Figure 3.13 shows an user interface of LX-ListQuestion online displaying the answers in the traditional list form.

The second way of visualizing answers is by using a wordcloud. In the case of wordcloud the most relevant answers appear in a larger and darker font color, the less relevant answers appear in minor font and lighter colors.

Figure 3.14 shows an user interface of LX-ListQuestion online using the wordcloud visualization.



Figure 3.14: LX-ListQuestion - online version interface - wordcloud Answers.

3.7 Summary

In this Chapter we presented our approach to answer natural language questions that require list of answers extracted from multiple documents from the Web. The design features and the tools that support of the system are also presented. We explained the development of the LX-ListQuestion system and the modules that composes the system were described in detail. Finally, we presented the LX-ListQuestion online version, which presents the answers in two ways: traditional list and wordcloud.

Answering Temporal List Questions

"Time structures our world, and the questions we ask reflect that" (Ahn *et al.*, 2006). Due to the continuous growth of the amount of information available, there is a need for automatic ways of searching for information. As selecting relevant information with temporal information is an important issue to the users, temporal information became an important research topic in NLP applications, such as Question Answering.

Regarding Temporal QA, UzZaman *et al.* (2012) wrote "The Web-based question answering systems are not discussed, since they rely heavily on answer redundancy instead of temporal reasoning". This statement shows that there is a lack of research that connects Web-Based and Temporal QA. In this dissertation we addressed this issue by joining the Web-based approach to collect all possible answers to the question with a shallow temporal processing approach to filter the answers based on the temporal restriction.

When asking a question that refers directly to a temporal expression, the answers need to be validated against the temporal constraints. To achieve such functionality, LX-ListQuestion was extended to identify temporal expressions and rely on this temporal expression to identify the answer satisfying the temporal restriction.

Our approach is based on extending the system described in Chapter 3. After finding all possible answers for the list question, it checks the temporal restriction in the same corpus retrieved from the Web by searching for temporal expressions in the free text. Only the answers that agree with the temporal restriction defined in the question will be selected to be presented to the user.

4. ANSWERING TEMPORAL LIST QUESTIONS

4.1 Outline

This chapter is organized as follows. Section 4.2 provides the background for Temporal Question Answering and presents the main concepts and challenges. We also describe how Temporal Questions are classified. Our approach is detailed in Section 4.3. We present the design features of LX-ListQuestion, the questions that are expected as input and we explain how we solve the time-range issue of the temporal expression. In Section 4.4 we describe in detail the architecture of the system. All modules are described and examples are discussed. Finally, a summary of this chapter is presented in Section 4.6.

4.2 Background

Given a text in natural language, understanding the temporal information requires anchoring and ordering the events of the text in time (UzZaman *et al.*, 2012). This task involves the extraction of temporal expressions (e.g., *1999*, *last year*, *5 hours*, *today*), events (e.g. *said*, *arrived*, *won*) and their temporal relations.

The *de facto* standard for classifying temporal relations is based on the work of Allen (1983) which defines thirteen types of relations: before, after, overlap, overlappedBy, start, startedBy, finished, finishedBy, during, contains, meet, metBy and simultaneous. Table 4.1 shows these relations.

Relation	Symbol	Symbol for Inverse	Pictorial Example
X before Y	<	>	XXX YYY
X equal Y	=	=	XXX YYY
X meets Y	m	mi	XXXYYY
X overlaps Y	o	oi	XXX YYY
X during Y	d	di	XXX YYYYYY
X starts Y	s	si	XXX YYYYYY
X finishes Y	f	fi	XXXX YYYYYY

Table 4.1: The relations defined by Allen (1983)

TimeML is a rich specification markup language used to annotate elements related with time (Saurí *et al.*, 2006). According to Pustejovsky *et al.* (2003), TimeML integrates three efforts in the semantic annotation of text: TimeML (i) systematically anchors event predicates to a broad range of temporally denoting expressions; (ii) provides a language for ordering event expressions in text relative to one another; and (iii) provides a semantics for underspecified temporal expressions, thereby allowing for a delayed interpretation. All elements related with time are annotated: time expressions (e.g. *1999*, *yesterday*, *January*, etc.), signals (e.g. *while*, *before*, *after*, etc.), events (e.g. *arrived*, *left*, *said*, etc.) and temporal relations (defined by Allen (1983)). Table 4.2 shows TimeML elements and their tags.

Element	TimeML Tag
Time Expression	<TIMEX3 >
Signal	<SIGNAL >
Event	<EVENT >
Temporal Relation	<TLINK >

Table 4.2: Tags of TimeML.

Figure 4.1 shows an example of TimeML annotation of the sentence: "John left 2 days before the attack".

```

John
<EVENT eid="e1" >
left
</EVENT >
<TIMEX3 tid="t1" >
2 days
</TIMEX3 >
<SIGNAL sid="s1" >
before
</SIGNAL >
the
<EVENT eid="e2" >
attack
</EVENT >
<TLINK eventInstanceID="e1" signalID="s1"
relatedToEvent="e2" relType="BEFORE" >
</TLINK >

```

Figure 4.1: Example of TimeML annotation.

4. ANSWERING TEMPORAL LIST QUESTIONS

4.2.1 Classification of Temporal Questions

Authors may define Temporal Questions in different ways . In Radev and Sundheim (2002), temporal questions are classified in two classes:

- (1) **Explicit temporal questions:** Questions that require a temporal expression as an answer.
 - a. When was Elvis Presley born?
 - b. At what time does the evening start?
 - c. In which century did Queen Elizabeth I reign?

- (2) **Implicit temporal questions:** Questions that have a temporal expression as a restriction. The answer is not a temporal expression.
 - a. Who was the president of the US in 1990?
 - b. Did world steel output increase in the nineties?

This classification by Radev and Sundheim (2002) is oversimplified. It was made in 2002, a time when the research on temporal questions was at its starting point. As the research on this topic progressed, the classification of temporal questions became more fine-grained, as the 16 classes of Harabagiu and Bejan (2005)¹ show:

1. **Factoid Temporal:** Questions that requires a date as an answer.
When did the Pope visit Poland?

2. **Time range:** Question with a single event and answer related with time (not a date).
How long did Iraq fight with Iran?

3. **Relative time range:** Question with a temporal range as a constraint. Answer is not related with time.
Where can I find research information in the Israeli Palestinian issues since 1991?

¹The examples were collected from the original paper of Harabagiu and Bejan (2005).

4. **Repetitive event:** Questions that requires a date as an answer related with a recurring event.
When does the *temporao* normally arrive in Brasil?
5. **Typical event:** Question with a typical event (e.g. on average) and the answer is not a date.
How long does it take on average to build a 500-room hotel in Las Vegas?
6. **Time anchored event:** Question with a temporal restriction (non time range) anchored with an event.
What important things happened in the year 1987?
7. **Events in time range:** Question with an event anchored with a temporal restriction.
What did George Bush do after the U.N. Security Council ordered a global embargo on trade with Iraq in August 90?
8. **Entities related to events/states changes:** Questions where an entity (e.g. world oil prices) changes over an event or a state.
What happened to world oil prices after the Iraq annexation of Kuwait ?
9. **Entity change in time period:** Questions where an entity changes over a time period.
I want to find pictures of presidents from the 1940-1949.
10. **Quantity change in time period:** Questions that require a quantity as an answer anchored to a temporal restriction.
How much did Las Vegas grow in population since 1980?
11. **Entity related to events at time stamp:** Questions where the entity is related with an event anchored in at time stamp.
Which two nations met in Washington on August 14, 1990 to discuss a naval blockade against Iraq?
12. **Age at time stamp:** Questions that require an age as an answer (type: “How old”) anchored to a temporal restriction.
How old was Michael Milken in January 1989?

4. ANSWERING TEMPORAL LIST QUESTIONS

13. **Comparative:** Comparative question anchored to a temporal restriction.
What is the difference between the teenager average weight today and in the 80's?
14. **Period of comparative/superlative attribute:** Questions with comparative or superlative attribute that requires a date as an answer.
When was the period of major growth in Las Vegas?
15. **Alternative temporal:** Alternative question with a temporal relation.
Did John Sununu resign before or after George Bush' ratings began to fail?
16. **Temporal relation:** Questions where the answer is anchored with a temporal relation.
Where did Michael Milken work while attending graduate school?

The classification of Harabagiu and Bejan (2005) is very elaborate. Besides having 16 types, the criteria for deciding how to classify a question vary between being related to the question and being related to the answer. Some types could be joined together because they are very similar.

More recently, Saquete *et al.* (2009) defines two major classes of Temporal Question: Simple and Complex, further divided into sub-classes as follow:

- **Simple:**

1. Questions that require a temporal expression as an answer and do not contain temporal expression.
When did man arrive on the moon?
2. Questions that require a temporal reasoning of the temporal expression contained in the question.
Who won the World Cup in 2010?

- **Complex:**

1. Questions with temporal expressions that contain more than one event related with a temporal signal (temporal signals are expressions like “before”, “after”, “between”, “when”).
What did George Bush do after the U.N. Security Council ordered a global embargo on trade with Iraq in August 90?

2. Questions without temporal expressions that contain more than one event related with a temporal signal.

Who was the president of the US when the AARP was founded?

The classification of Saquete *et al.* (2009) is directed towards the research developed by them which focuses on Complex Temporal Questions.

4.2.2 Challenges in Temporal Question Answering

Temporal Question Answering raises some specific challenges that are not found in general QA. Pustejovsky *et al.* (2002) bring to light some challenges in Temporal Question Answering. The authors address four research challenges:

1. Time stamping of events — identifying an event and anchoring it in time.
2. Ordering events with respect to each other — relating more than one event in terms of precedence, overlap and inclusion.
3. Reasoning about the ramifications of an event — identifying changes by virtue of an event.
4. Reasoning about the persistence of an event — identifying how long does an event or the outcome of an event persist.

In this dissertation we address an approach to tackle the first challenge, finding answers anchoring in the time expressed in the question.

4.2.3 Related Work

The seminal work on Temporal Question Answering was related to the development of annotated corpora (Radev and Sundheim, 2002), (Schilder and Habel, 2003) and (Ahn *et al.*, 2006). The idea was reasoning over the corpora to find answers related with time.

The overview of the state-of-the-art shows a lot of effort on Complex Temporal Questions. We found no predominant approach, since each researcher chose a different path depending on the type of temporal questions. Harabagiu and Bejan (2005) developed a

4. ANSWERING TEMPORAL LIST QUESTIONS

methodology that use annotation produced by TimeML to generate a template to answer temporal questions.

Schockaert *et al.* (2006) presents an algorithm based on an algebra to deduce temporal knowledge to answer temporal questions. The work presented in Saquete *et al.* (2009) developed an approach based on decomposing the question into simple questions, according to the temporal relations expressed in the original question and at the end of processing, the answers are recomposed to find the correct answers.

There are still some works on Temporal Question Answering using the Web as information source. Paşca (2008) answers only simple factoid temporal questions extracting information from snippets and storing them in repositories. Tao *et al.* (2010) explore the semantic web using an OWL ontology to answer complex temporal questions. The system works over closed corpora over a specific domain, in this case, a clinical corpus. The system developed by Yahya *et al.* (2013b) uses structured knowledge bases available on the web to search simple factoid and temporally restricted temporal questions. See more about related work in Section 2.7 in Chapter 2.

4.3 Approach

In our research we did not find other systems that deal with list temporal questions. Besides tackling this under-researched topic, our approach is innovative in the way it join the Web-based approach to collect all possible answers to the question with a shallow temporal processing approach to filter the answers based on the temporal restriction.

According to Harabagiu and Bejan (2005), processing questions that involve temporal restriction relies on (1) the recognition of events or entities that participate in them; (2) the relative ordering of events and entities in the corpus and (3) the identification of temporal expressions and the relation with the expected answers. Following this perspective, LX-ListQuestion was extended on three main processing stages:

1. Question processing for interpreting the question and identification of the temporal restriction. After identifying the temporal restriction, the system defines the boundaries of the time-range of the question.
2. Document processing for selecting the relevant information from the Web corpus. This processing is very similar to the one mentioned earlier in Section 3.5.2.1.

3. Answer processing for selecting the right answers for the question respecting the temporal constraint on the question.

4.3.1 Design Features

LX-ListQuestion with special attention to temporal questions was built using as starting point the system described in Chapter 3. Recall the design features described in Section 3.4:

- Exploits redundancy to find all answers to the List Question;
- Compiles and extracts the answers from multiple documents;
- Collects at run-time the documents from Web using a search engine;
- Provides answers in real time without resorting to previously stored information.

To these, we add two more:

- Transforms the temporal information contained in the question into a temporal constraint;
- Filters the answers using temporal constraints;

4.3.2 Expected Input

Our system expects as input questions in Portuguese that require a list of answers with temporal restriction. The temporal restrictions which the system handles are years and centuries. LX-ListQuestion answers questions which expect named entities as an answer. Table 4.3 shows examples of List Temporal Questions which LX-ListQuestion is capable to answer.

Examples:
Quais eram os partidos políticos existentes antes de 1964?
Quais são os reis de Portugal entre 1500 e 1700?
Quem ganhou o premio nobel entre 1900 a 1920?
Quais são os livros da Danielle Steel na década de 80?
Que países boicotaram os Jogos Olímpicos de 1980?
Cite os países que disputaram território com o Brasil antes de 1900?

Table 4.3: Examples of expected input of LX-ListQuestion.

4. ANSWERING TEMPORAL LIST QUESTIONS

4.3.3 Classification of Temporal List Questions

Since none of the classifications mentioned in Section 4.2.1 include categories specific to Temporal List Question, we chose to adapt the classifications described by Radev and Sundheim (2002) and Saquete *et al.* (2009) to classify this type of question. We assume three types of temporal questions: simple, complex and temporally restricted.

- **Simple:** Explicit temporal question that requires a temporal expression as an answer.

In which years Brazil won the World Cup?

- **Complex:** Questions with more than one event related by a temporal expression. The temporal expression establishes the order between the events in the question. The answer is not a temporal expression.

Which movies did Sam Raimi direct after Army of Darkness?

- **Temporally Restricted:** Questions that have a temporal expression as a restriction. The answer is not a temporal expression.

Which films Steven Spielberg directed in the 80's?

Name all songs of Bruce Springsteen released between 1980 and 1990.

The focus of this dissertation is to answer List Question of the Temporally Restricted type. Although according to our classification Simple Temporal List Questions are possible, we did not find this kind of question in our corpus. It appears more easily in Factoid Questions (e.g. "When" question type). We assume that this type of temporal question is rare in list questions and, as such, excluded it from our study.

Regarding Complex Temporal Questions, the review of the state-of-the-art shows that it is necessary to have access to corpora annotated with temporal relations in order to work on this type of questions. Given that such resources for Portuguese are still incipient, and to allow us to better focus our effort, we also exclude this type of questions from our study.

4.3.4 Solving Time-Range

Answering Temporally Restricted Questions is a non-trivial task. This topic was approached by Moldovan *et al.* (2005) and Schockaert *et al.* (2006). Based on an analysis of corpora we identify three classes of temporal restriction: (1) Temporal restriction in an absolute time, (2) Temporal restriction with a relative reference, and (3) Temporal restriction with a vague reference.

4.3.4.1 Temporal Restriction in an Absolute Time

Questions with temporal restriction in an absolute time are very common temporal questions. In terms of question processing it is easy to identify this type of question. In this case we identified the temporal expression in the question and the temporal expression is the temporal restriction. For absolute time, a temporal restriction can be single (e.g. a year) or multiple (e.g. list of years). Table 4.4 shows examples.

Type	Question	Time Restriction
Single	O que tocava nas rádios no ano 2010?	2010
Single	Quais foram os fatos importantes ocorridos no ano 2000?	2000
Multiple	Quais os movimentos trabalhistas que ocorreram em 1980, 1990 e 2000?	1980, 1990, 2000
Multiple	Que moeda era vigente no Brasil em 1967, 1970, 1986, 1989, 1990, 1993 e 1994?	1967, 1970, 1986, 1989, 1990, 1993, 1994

Table 4.4: Examples of questions with temporal restriction in an absolute time.

4.3.4.2 Temporal Restriction with a Relative Reference

Questions with temporal restriction with a relative reference require careful processing. The system finds the relative references and solves the time-range by anchoring these temporal references in a calendar year. The relative references can be located using the following closed set of expressions: *a partir de*, *depois de*, *até*, *antes de*, *últimos*, *entre*, *anos*, *década* and *século*, (ENG *starting at*, *after*, *until*, *before*, *the last*, *between*, *years*, *decade* and *century*). Table 4.5 shows examples assuming 2014 as the current year.

4. ANSWERING TEMPORAL LIST QUESTIONS

Question	Time Expression	Time-Range
Quais foram os conflitos que atingiram o Afeganistão a partir de 1970?	a partir de 1970	1970 - 2014
Quais os países europeus surgidos depois de 1989?	depois de 1989	1989 - 2014
Cite filmes de comédia até 2005	até 2005	[...] - 2005
Quais eram os títulos do Corinthians antes de 2000?	antes de 2000	[...] - 2000
Quais foram as novelas brasileiras dos últimos 5 anos?	últimos 5 anos	2009 - 2014
Quem ganhou o premio nobel entre 1900 a 1920?	entre 1900 e 1920	1900 - 1920
Liste as bandas famosas dos anos 80.	anos 80	1980 - 1989
Quais são os clubes campeões mundiais na década de 1950?	década de 1950	1950 - 1959
Liste as empresas fundadas no século XX	século XX	1901 - 2000

Table 4.5: Examples of questions with temporal restriction with a relative reference

4.3.4.3 Temporal Restriction with a Vague Reference

Temporal reasoning with a vague reference is further complicated by the fact that the time span cannot be accurately captured by an interval with well-defined boundaries (Schockaert *et al.*, 2006). In our approach, when the system identifies a vague temporal expression, this temporal expression is used as a keyword. The list of vague temporal expressions was compiled on the basis of an analysis of a corpus of questions. The temporal expressions are: *período*, *era*, *época* and *tempo*. Examples are show in Table 4.6.

Question	Time Expression
Liste as grandes cidades do período romano.	período romano
Apresente três características dos jogos olímpicos da era moderna.	era moderna
Cite os grandes portos da época dos Grandes Descobrimentos.	época dos Grandes Descobrimentos
Qual o nome do instrumento usado para medir o tempo na era antiga?	era antiga

Table 4.6: Examples of questions with temporal restriction with a vague reference

4.4 Architecture

The previous Sections described the background on Temporal Questions Answering and discussed our approach using the shallow temporal processing to answer Temporal List Question. This Section describes all changes implemented in LX-ListQuestion to handle this type of question. All changes implemented in each module of the system are described in detail.

Figure 4.2 summarizes the process that join the redundancy and the shallow temporal processing.

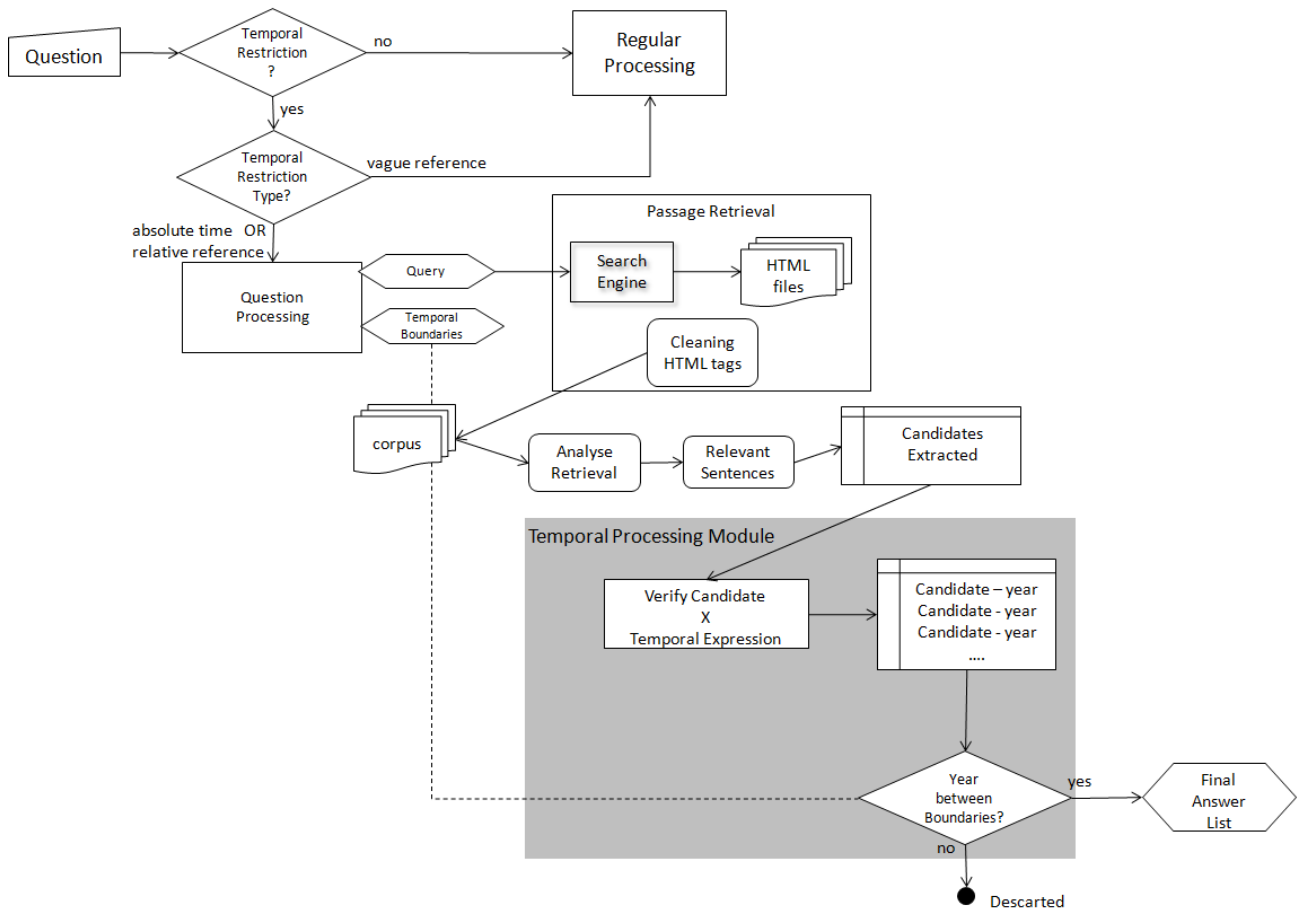


Figure 4.2: Summary of the processing of temporal list questions.

4.4.1 Question Processing Module

Question Processing Module underwent most of the changes. Besides all procedures already described in Section 3.5.1, the module is also responsible for: (i) identifying the temporal expression and (ii) defining the temporal boundaries (if necessary) to solve the time-range;

4. ANSWERING TEMPORAL LIST QUESTIONS

4.4.1.1 Identifying the Temporal Expression

The process of identification of temporal expression is based on hand-build patterns.

Temporal restriction in an absolute time

Table 4.7 shows the patterns built for the identification of temporal restriction in an absolute time. Note that for temporal restriction in an absolute time the time expression and the time restriction are the same.

Pattern	Question	Temporal Restriction
(ANO)	Liste os filmes de 2012.	2012
(ANO), (ANO)*	Liste os filmes de 1989, 1990 e 1994.	1989, 1990, 1994

Table 4.7: Examples of patterns to identify temporal restriction in an absolute time

Temporal restriction with a relative reference

Table 4.8 shows the patterns built to identify the temporal restriction with a relative reference. Besides being identified, temporal expressions are normalized into temporal boundaries: Upper and Lower.

Pattern	Question	Temporal Boundaries	
		Lower	Upper
anos (Numeral)	Liste os filmes dos anos 90	1990	1999
década (PREP) (Numeral)	Liste os filmes da década de 90 .	1990	1999
século (Letras/Numeral)	Liste as empresas fundadas no século XX . Liste as empresas fundadas no século vinte .	1901 1901	2000 2000
a partir (PREP) (ANO)	Liste os filmes a partir de 1990 .	1990	(Current Year)
a partir do século (Letras/Numeral)	Liste as empresas fundadas a partir do século XX .	1901	(Current Year)
depois (PREP) (ANO)	Liste os filmes depois de 1990 .	1990	(Current Year)
depois do século (Letras/Numeral)	Liste as empresas fundadas depois do século XX .	1901	(Current Year)
até (ANO)	Liste os filmes até 1990 .	[...]	1990
até o século (Letras/Numeral)	Liste as empresas fundadas até o século XX .	[...]	2000
antes (PREP) (ANO)	Liste os filmes antes de 1990 .	[...]	1990
antes do século (Letras/Numeral)	Liste as empresas fundadas antes do século XX .	[...]	1900
entre (ANO) e (ANO)	Liste os filmes entre 1970 e 1990 .	1970	1990
entre os séculos (Letras/Numeral) e (Letras/Numeral)	Liste as empresas fundadas entre os séculos XIX e XX .	1801	2000
últimos (Numeral) anos	Liste os filmes dos últimos 10 anos .	(Current Year) -10 years	(Current Year)

Table 4.8: Examples of patterns to identify temporal restriction with a relative reference

4. ANSWERING TEMPORAL LIST QUESTIONS

Temporal restriction with a vague reference

Table 4.9 shows the patterns built to identify the temporal restriction with a vague reference. For this type of restriction, when a temporal expression is identified, LX-ListQuestion performs normal processing using the temporal expression as keyword. Details are described in Chapter 3.

Pattern	Question
(PREP) período	Liste as grandes cidades do período romano .
(PREP) era	Liste os dinossauros da era paleolítica .
(PREP) época	Liste os grandes portos da época dos Grandes Descobrimentos .
(PREP) tempo	Liste os Quilombos no tempo da escravidão .

Table 4.9: Examples of temporal questions without explicit datetime mark.

4.4.1.2 Defining the Temporal Boundaries

The algorithm that processed the temporal expression and defines the temporal boundaries (Upper and Lower) is only applied on Questions with a temporal relative reference. To do this we use a mapping of the expression to a time interval. Table 4.10 shows an example:

	Time Range	
anos 20	1920	1929
anos vinte		
anos 30	1930	1939
anos trinta		
...		
anos 2000	2000	2009
anos dois mil		

	Time Range	
século I	1	100
século um		
século II	101	2000
século dois		
...		
século XXI	2001	2100
século vinte e um		

Table 4.10: Examples of mapping of the expression to a time interval.

4.4.1.3 Generating the Query

The query generation was already explained in Section 3.5.1.2. As mention before, the query is composed by only some part-of-speech categories: proper names, common nouns, verbs and adjectives. The articles, prepositions and pronouns are discarded. Specially for Temporal Questions, the system omit the temporal expression to compose the query. Table 4.11 shows some examples of Temporal List Question and their respective query generated by our system:

Temporal List Question	Query
Quais são os livros da Danielle Steel na década de 80?	livros Daniella Steel
Quais são os filmes de comédia de 2011 e 2012 ?	filmes comédia
Quais são os reis de Portugal entre 1500 e 1700?	reis Portugal
Nomeie artistas contemporaneos depois de 1950?	artistas contemporaneos

Table 4.11: Examples of questions and query

4.4.2 Document Processing Module

The document processing module is responsible for collecting the documents from the Web using the query generated by the previous module. The process is the same described in Section 3.5.2.1. After collecting the documents, the relevant sentences are set apart for identifying the candidate answers.

In this stage, the system will select the relevant information of the documents according with the relevance with the questions: All relevant sentences are set apart (Figure 4.3).

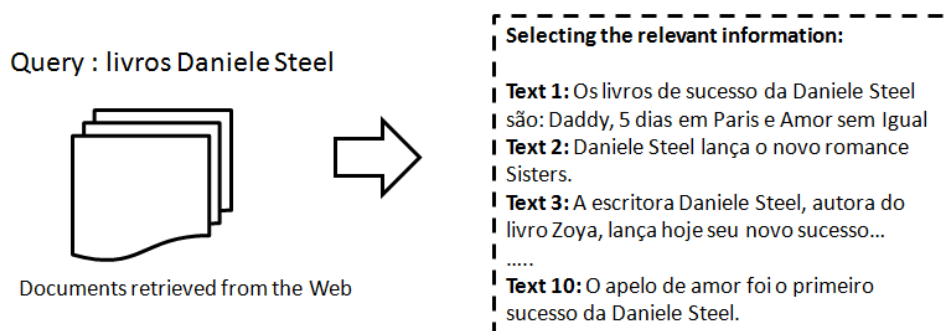


Figure 4.3: Document analysis for temporal list questions.

4. ANSWERING TEMPORAL LIST QUESTIONS

4.4.3 Answer Processing Module

The Answer Processing Module aims at identifying and extracting relevant candidates and building the final list of answers. This module was extended with a module for Temporal Processing that is responsible for selecting the correct answers using the temporal constraints as a filter. The process of extracting candidate answers and building the list of answers is explained below.

Extracting Candidate Answers

In this step it is very important to have the maximum number of candidates for the system to start the processing. The candidates are extracted from the sentences previously processed by the analysis of the retrieved documents. The process of extracting candidates is the same as previously explained in Section 3.5.3.1. Examples of the extracted candidates are highlighted below in the sentences in Figure 4.4.

Relevant Sentences:

Os livros de sucesso da Daniele Steel são: **Daddy, 5 dias em Paris** e **Amor sem Igual**.
Daniele Steel lança o novo romance **Sisters**.
A escritora Daniele Steel, autora do livro **Zoya**, lança hoje seu novo sucesso.
O apelo de amor foi o primeiro sucesso da Daniele Steel.
Na década de 80 uns dos mais famosos livros da Daniele foi **Zoya**.
Mais um grande livro da escritora Daniele Steel, **O anel de noivado** chega hoje a venda.
Casa forte, um dos livros mais vendidos desta década conta uma história fascinante.
A autora do livro **Álbum de Família**, Danielle Steel, autografa hoje seu livro.

Figure 4.4: Extracting candidate answers for temporal list question.

Building the Answer List with Shallow Temporal Processing

Shallow Temporal Processing is responsible for building the answer list to Temporal List questions. This process goes through two steps:

- The first step is to identify all temporal expression that appears in the same sentence for all candidates. For each candidate, we extract from the full set of documents all temporal expression that co-occur in the same sentence with that candidate. Figure 4.5 shows an example:

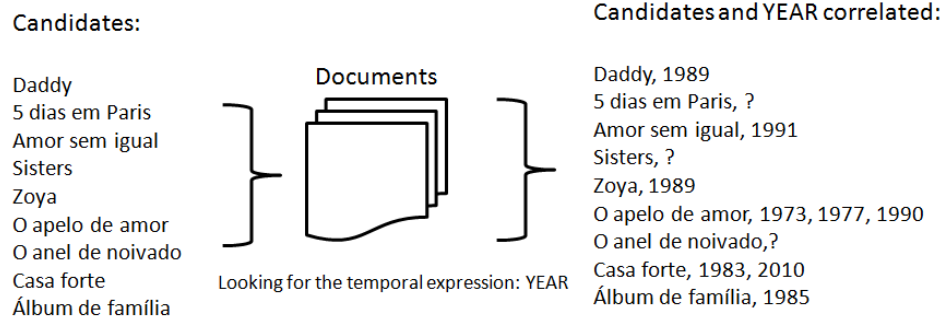


Figure 4.5: Finding co-occurring temporal expressions.

- The second step enforces the temporal restriction that appears in the question. For candidates with co-occurring temporal expressions, each expression is checked against the temporal boundaries. If any of co-occurring temporal expressions lies within the boundary, the candidate is classified as *ACCEPTED* and enters to the final list of answers. Otherwise, the candidate is classified as *REJECTED*. Table 4.12 shows examples of candidates with their temporal expressions found in the corpus.

Candidate	Temporal Expression found in the document corpus	ACCEPTED OR REJECTED
Daddy	1989	ACCEPTED
5 dias em Paris	[no temporal expression found]	REJECTED
Amor sem igual	1991	REJECTED
Sister	[no temporal expression found]	REJECTED
Zoya	1989	ACCEPTED
O apelo do amor	1973,1977, 1990	REJECTED
O anel de noivado	[no temporal expression found]	REJECTED
Casa forte	1983	ACCEPTED
Álbum de família	1985	ACCEPTED

Table 4.12: Building the answer list for temporal list questions.

Following the information in Table 4.12, the candidates: “Daddy”, “Zoya”, “Casa forte” and “Álbum de família”, were *ACCEPTED* since there is some temporal expression that is within the established time boundaries. The remaining candidates were *REJECTED* for different reasons: “5 dias em Paris”, “Sister” and “O anel de noivado” were *REJECTED* since no temporal expression was found associated with

4. ANSWERING TEMPORAL LIST QUESTIONS

them; “Amor sem igual” and “O apelo do amor” were REJECTED since all co-occurring temporal expressions are outside the boundaries of the temporal restriction given by the question.

4.5 LX-ListQuestion: QA system to List Question with Temporal Restrictors

The temporal processing module was integrated into the Web-based QA system, enabling it to also answer questions with temporal restrictors. The architecture and user interface of the system were already presented in Section 3.6. Note that candidate relevance is not taken into account by the temporal processing module. As such, the wordcloud view shows all answers using the same font size, while the list view, instead of ordering answers by relevance, orders them chronologically. An example screenshot may be seen in Figure 4.6.



Figure 4.6: LX-Listquestion online QA system - list question with temporal restrictors

4.6 Summary

This chapter had two main goals. The first one was to provide a review of the current state-of-the-art on Temporal Question Answering and situate our system within the field. Based on what we have gathered, we can say that QA for Temporal List Question is a under-research topic. In this regard, the current work addresses a topic that was lacking specific research.

The second goal was to present our innovative approach that combines redundancy and temporal shallow processing. The process is executed in two steps. In the first step, redundancy is used to collect all potential answers to the List Question. In the second step, temporal shallow processing is responsible for verifying if each potential answer satisfies the temporal restriction given by the question.

The system not only answers questions with an absolute time restriction (e.g. “*Que países boicotaram os Jogos Olímpicos de 1980?*” EN:“*Which countries boycotted the Olympics Games in 1980?*”), but also answers questions with a relative time restriction (e.g. “*Cite os livros da Daniele Steel da década de 80.*” EN:“*Name all books of Daniele Steel in the 80’s*”) using an algorithm that enforces the time-range boundaries. The result of this is the extension of the LX-ListQuestion Web-based QA system for List Questions described in Chapter 3 with functionality that enables it to answer List questions with a temporal restriction.

5

Evaluation

The evaluation of a Question Answering system is a challenging task. It requires a dataset composed by questions and respective answers. In this Chapter, we use different question datasets to test our system in different scenarios. For instance, the Páxico Question dataset provides questions focusing on issues related to Portuguese culture, while the QALD question dataset provides a set of questions with an international scope.

LX-ListQuestion is a Web-based QA System that focuses on answering list questions whose answers are extracted and composed from several documents retrieved from the Web, as already described in Chapter 3. The first part of this Chapter is dedicated to evaluating and discussing the results obtained by our system. In order to assess the positioning of our system in the state-of-the-art we compare LX-ListQuestion with four other QA systems. For the comparison, the results were analyzed in two ways: (i) the quantitative evaluation of answers provides recall, precision and F-measure and (ii) the question coverage that indicate the usefulness of the system to the user counting the number of questions for which the system provides at least one correct answer.

In the second part of this Chapter, we consider the Temporal Processing Module connected to LX-ListQuestion. Our approach to answer Temporal List Questions is based on extended LX-ListQuestion system described in Chapter 4. After finding all possible answers for the list question, it checks the temporal restriction in the same corpus retrieved from the Web by searching for temporal expressions in the free text. In addition to the evaluation of answering Temporal List Questions, we also compare the performance of our system with other QA systems.

5. EVALUATION

5.1 Outline

Section 5.2 will explain our algorithm of automatic evaluation to facilitate the evaluation task. To perform the experiments in this Chapter, we used two Question Datasets described in Section 5.3. Section 5.4 presents the evaluation of LX-ListQuestion in different ways. First we present all correct answers found in corpora of different size. Afterwards, we test the system using different threshold parameters. We also present the evaluation of LX-ListQuestion using four different setups in order to test the efficiency of our approach.

Section 5.5 compares the results against four different QA Systems: RapPortagico, a off-line QA System; XisQuê, a Web-based QA system for Portuguese; START, another Web-based QA system for English; and WolframAlpha, a knowledge engine. Each system is presented and its design features are compared with the design features of LX-ListQuestion. In order to assess the positioning of our system in the state-of-the-art, our evaluation has two components: the quantitative evaluation of answers and the question coverage evaluation.

The Temporal Processing module of LX-ListQuestion is evaluated in Section 5.6. We test our approach by applying different temporal restrictors for the same base question. Afterwards, we compare the results for Temporal List Questions against RapPortagico and XisQuê. Finally, Section 5.7 concludes with a summary and some final remarks.

5.2 Automatic Evaluation

Most of our experiments were based on the Question Dataset of the Págico Competition. This competition built the list of answers as list of links to Wikipedia web-pages. Figure 5.1 shows an excerpt of the correct answers of question id: Pagico_004 - Mulheres violoncelistas de língua portuguesa.

```
...
Pagico_004 pt/c/a/r/Carmen_Monarcha.19d49b.xml
Pagico_004 pt/d/e/n/Denise_Emmer.6b270a.xml
Pagico_004 pt/g/u/i/Guilhermina_Suggia.1a7652.xml
...
```

Figure 5.1: Original answers given by Págico competition.

...
Pagico_004 Carmen Monarcha
Pagico_004 Denise Emmer
Pagico_004 Guilhermina Suggia
...

Figure 5.2: Answers given by Págico competition after cleaning.

LX-ListQuestion seeks for answers within documents retrieved from Web. To undertake its assessment, we assume that the name of the web-page is the correct answer to the question. We manually clean the answers as showing in Table 5.2. After cleaning the answers in the original file given by Págico Competition, we remain only with a answer in text format.

To facilitate this task we need to compare the reference list of answers with the list given by the system in an automatic way. Note that there are several cases where the system answers can be different from the ones in the reference list and yet being correct, due to many factors like spelling differences, omissions of part of proper-names, abbreviations and other. Figure 5.3 shows some examples of answers that are different but still correct.

Answer (A)	Answer (B)
Pisa	Província de Pisa
Etiópia	Ethiópia
Keith Charles Flint	Keith Flint
Nossa Senhora do Rosário de São Bendito	N. S. do Rosário de São Benedito

Figure 5.3: Variation in correct answers

Considering these cases, it is necessary to create an algorithm that instead of a strict string matching, uses a more relaxed process to smooth the correctness of their answers.

We implemented an algorithm of automatic evaluation based on the overlap of common words and on Levenshtein Distance¹. Each word in the candidate answer is compared with the reference answer from Págico. If the Levenshtein Distance is less than 3, the words are considered similar enough to match. The candidate answer receives a score based on the number of common words over the set of words in the reference and system answers.

Our automatic evaluation allows marking the answers with a certain degree of certainty. Figure 5.4 shows some examples of the automatic evaluation.

¹The Levenshtein distance is a string metric for measuring the difference between two sequences. Informally, the Levenshtein distance between two words is the minimum number of single-character edits (i.e. insertions, deletions or substitutions) required to change one word into the other.

5. EVALUATION

Correct Answer	Answer by LX-ListQuestion	Score	Accept
ilha mocambique	ilha mozambique	1.0	YES
malanje	malange	1.0	YES
baltasar lopes silva	baltasar lopes	0.67	YES
sao paulo cidade	sao paulo	0.67	YES
igreja nossa senhora rosario sao benedito rio janeiro	nossa senhora rosario sao benedito	0.62	YES
praia cabedelo viana castelo	praia cabedelo	0.50	YES
manuel novas	manuel ferreira	0.33	NO
bazaruto	hotel pestana bazaruto lodge	0.33	NO
luena angola	angola portal	0.33	NO
joao branco nuncio	francisco nuncio	0.25	NO
sao joao estoril	monte estoril	0.25	NO
igreja nossa senhora rosario sao benedito rio janeiro	capela nossa senhora	0.22	NO

Figure 5.4: Relaxed candidate matching in the automatic evaluation.

5.3 Question Dataset

Two Question Datasets were used to perform the experiments in this Chapter:

Question Dataset of Páxico Competition The whole dataset is composed by 150 questions about Lusophony extracted from the Portuguese Wikipedia¹. The questions are about Geography, History, Politics, Science and others. The main criteria when the corpus was built was to ensure that the search for the answers is non-trivial, and that the answers are spread throughout multiple documents (Freitas, 2012). For the experiments, we use a subset of 30 questions whose expected answer type is Person or Location. We pick these two types since they are the ones more accurately assigned by the underlying LX-NER tool. Note, however, that our approach is not intrinsically limited to only these types. The subset of questions used in the experiments appears in detail in Figure 5.1. All questions require a list of answers, amounting to 340 answers in total, and on average, around 11 answers for each question.

Question Dataset of QALD The whole dataset² is composed by 200 questions in English. The dataset is composed by Factoid, Definition, Boolean and List Questions. We manually annotated each question with the expected answer type. For those questions we randomly select 10 questions of List type, which were translated into Portuguese, to compose the subset used in the Experiments as detailed in Table 5.2.

¹ All information can be found in <http://www.linguateca.pt/Cartola/> - last access on December, 1 2014.

² QALD is a series of evaluation campaigns on multilingual question answering over linked data, currently part of the Question Answering lab at CLEF. All information can be found in <http://greententacle.techfak.uni-bielefeld.de/cunger/qald/> - last access on December, 1 2014.

5.3 Question Dataset

Pagico_004	Mulheres violoncelistas de língua portuguesa. <i>Portuguese-language female cellists.</i>
Pagico_053	Parques do Rio de Janeiro que têm cachoeiras. <i>Parks with waterfalls in Rio de Janeiro.</i>
Pagico_054	Igrejas do Rio de Janeiro construídas por irmandades ou confrarias de negros. <i>Churches in Rio de Janeiro built by black religious brotherhoods or fraternities.</i>
Pagico_058	Países que venceram a Copa do Mundo em uma disputa de pênaltis. <i>Countries that won the World Cup by penalty shootouts.</i>
Pagico_059	Jogadores de basquetebol brasileiros que jogam ou jogaram em campeonatos da NBA. <i>Brazilian basketball players that play or have played in the NBA.</i>
Pagico_062	Praias de Portugal boas para a prática de surf. <i>Good Portuguese beaches for surfing.</i>
Pagico_063	Estudiosos da música indígena brasileira. <i>Scholars of indigenous Brazilian music.</i>
Pagico_085	Destinos turísticos do Brasil cuja temperatura no Inverno pode ser negativa. <i>Brazilian tourist destinations where the winter temperature can be negative.</i>
Pagico_086	Compositoras brasileiras de samba. <i>Brazilian samba songwriters.</i>
Pagico_088	Cidades portuguesas que têm festivais medievais. <i>Portuguese cities that have medieval festivals.</i>
Pagico_091	Estados fronteiriços de Moçambique. <i>Mozambican border-states.</i>
Pagico_092	Cidades que fizeram parte do domínio português na Índia. <i>Cities that were part of the Portuguese Empire in India.</i>
Pagico_094	Parques nacionais de Moçambique. <i>Mozambican national parks.</i>
Pagico_097	Escritores cabo-verdianos com obra publicada em crioulo. <i>Cape Verdean writers with published work in creole.</i>
Pagico_100	Ilhas de Moçambique. <i>Mozambican islands.</i>
Pagico_104	Pesquisadores do folclore brasileiro. <i>Brazilian folklore researchers.</i>
Pagico_106	Vice-reis da Índia Portuguesa. <i>Viceroy of Portuguese India.</i>
Pagico_108	Jogadores de futebol nascidos em Cabo Verde que representaram a seleção portuguesa. <i>Football players born in Cape Verde who have represented the Portuguese national team.</i>
Pagico_109	Candidatos a alguma das eleições presidenciais na Guiné-Bissau. <i>Candidates for any presidential elections in Guinea-Bissau.</i>
Pagico_111	Padres católicos que estão ou estiveram ativos em Timor. <i>Catholic priests who are or were active in Timor.</i>
Pagico_112	Capitais das províncias de Angola. <i>The capitals of Angolan provinces.</i>
Pagico_116	Escritores lusófonos que passaram temporadas na prisão. <i>Lusophone writers who spent time in prison.</i>

5. EVALUATION

Pagico_118	Escritores moçambicanos que receberam o Prémio Camões. <i>Mozambican writers who have received The Camões Prize.</i>
Pagico_124	Cabo-verdianos que participaram na guerra colonial na Guiné. <i>Cape Verdeans who participated in the colonial war in Guinea.</i>
Pagico_128	Escritores portugueses que tenham vivido em Macau. <i>Portuguese writers who have lived in Macau.</i>
Pagico_132	Deputados da FRELIMO. <i>FRELIMO's deputies.</i>
Pagico_133	Futebolistas do Petro de Luanda. <i>Petro de Luanda players.</i>
Pagico_140	Cidades lusófonas conhecidas pelo seu Carnaval. <i>Lusophone cities known for their carnival celebrations.</i>
Pagico_149	Arquitetos de países lusófonos com obras em países estrangeiros na América do Norte e na Europa. <i>Architects from lusophone countries with works in foreign countries in North America and Europe.</i>
Pagico_153	Toureiros a cavalo de países lusófonos com carreira internacional. <i>Internationally-known bullfighters on horseback from lusophone countries.</i>

Table 5.1: Subset of question dataset - Págico Competition

QALD_010	<i>In which country does the Nile start?</i> Em qual país começa o Rio Nilo?
QALD_028	<i>Give me all communist countries.</i> Liste todos países comunistas.
QALD_032	<i>Which countries adopted the Euro?</i> Quais são os países que adotaram o euro?
QALD_036	<i>Through which countries does the Yenisei river flow?</i> Por quais países o rio Yenisei corre?
QALD_062	<i>Who created Wikipedia?</i> Quais são os criadores da Wikipedia?
QALD_074	<i>Which capitals in Europe were host cities of the summer Olympic games?</i> Quais as capitais na Europa que hospedaram o Jogos Olímpicos de Verão?
QALD_114	<i>Give me all members of Prodigy.</i> Cite todos membros do Prodigy.
QALD_141	<i>Who founded Intel?</i> Quais são os fundadores da Intel?
QALD_155	<i>Which Greek goddesses dwelt on Mount Olympus?</i> Quais Deusas gregas moravam no Monte Olimpo?
QALD_176	<i>List the children of Margaret Thatcher.</i> Liste os filhos de Margaret Thatcher.

Table 5.2: Subset of question dataset - QALD

5.4 List Questions: Evaluation

In this section we evaluate the LX-ListQuestion system in different ways. We begin by assessing the impact of the number of retrieved documents on recall. Following this, we test how different values for threshold parameters affect the filtering of candidates. The evaluation proceed with testing different setups for the system.

5.4.1 The role of Document Retrieval

In this Section we assess the role of the Document Retrieval module. Our goal is to verify how the number of correct answers varies when the amount of documents retrieved from the Web is changed.

Experimental Setup: The question dataset used was presented in Table 5.1. We evaluate a set of List Questions using 5, 10, 15 and 20 documents retrieved from the Web to build a document corpus. The system picks the relevant sentences and extracts all candidates without applying any threshold.

Figure 5.5 illustrates the number of distinct correct answers found when the number of documents retrieved from the Web is changed.

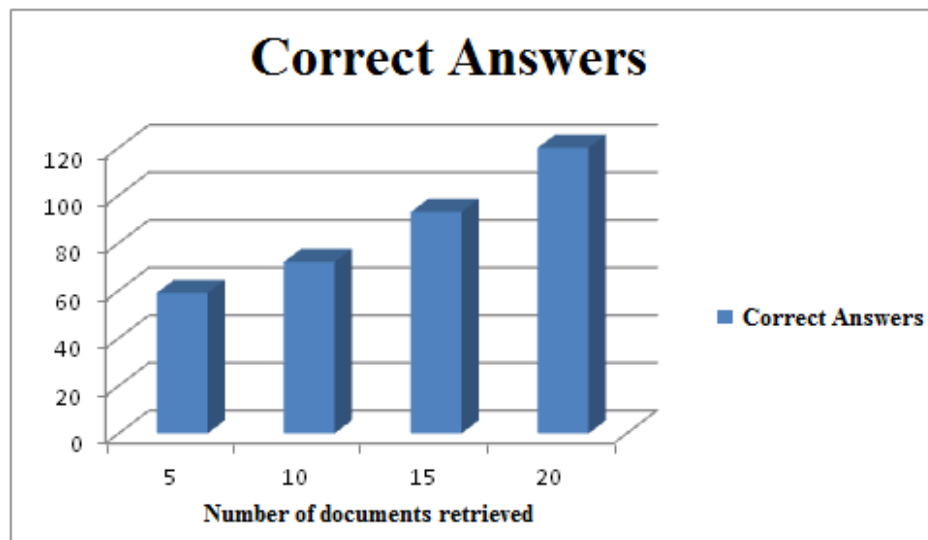


Figure 5.5: Correct answers found on the corpus.

5. EVALUATION

The total number of correct answers found is 59 (when 5 documents are retrieved), 72 (for 10 documents retrieved), 93 (for 15 documents retrieved) and 120 (for 20 documents retrieved). When we increase the number of documents, the number of correct answers increases as well. However, this increase in the number of available correct answers comes with an increase in the amount of candidates, which brings noise into the process.

Table 5.3 shows the number of correct answers and the total of candidates for each corpus size. Note that, regardless of the number of retrieved documents, less than 2% of the candidates found in the corpus are correct answers. Given that none of the scenarios stand out as being less noisy than the others, we opt by setting the number of retrieved documents at 10, since this value is commonly pointed out as providing the best results (Dumais *et al.*, 2002).

	5 documents	10 documents	15 documents	20 documents
#CorrectAnswers	59	72	93	120
#Candidates	8138	12369	17038	23758

Table 5.3: Corpus composition

5.4.2 Evaluation using Different Threshold Parameters

Our Bipartite List Approach exploits redundancy to find all answers to the List questions, and uses their frequency as a factor to select the correct answer. We built two lists with the candidates: Premium List (PL) and Work List (WL). In each list, the candidates that appear repeated are grouped together and their frequency is calculated. Following this, two frequency thresholds are calculated. A more relaxed threshold, named TP, is used to filter the candidates of Premium List and a more stringent one, named TW, to filter the candidates of Work List. These thresholds are bounded by different parameters (for full detail see Section 3.5.3.2). In this Section we aim to verify how many correct answers LX-ListQuestion finds using different values for Threshold Parameters.

Experimental Setup: The question dataset used was presented in Table 5.1. The Documents Corpus setup is the one that uses 10 retrieved documents, as results from the previous experiment. Given that LX-ListQuestion is a Web-based system, we choose to freeze the document corpus to ensure the repeatability of the experiments. The system uses two different thresholds, TP and TW. Both thresholds are parameterizable. In this experiment we test how the F-measure is affected by different parameters for TP and TW.

5.4 List Questions: Evaluation

	TW(1)	TW(2)	TW(3)	TW(4)	TW(5)	TW(6)	TW(7)	TW(8)	TW(9)	TW(10)
TP(1)	0.091	0.092	0.085	0.083	0.089	0.093	0.091	0.096	0.093	0.086
TP(2)	0.087	0.088	0.082	0.080	0.086	0.090	0.088	0.090	0.090	0.084
TP(3)	0.095	0.096	0.089	0.088	0.093	0.096	0.094	0.096	0.096	0.090
TP(4)	0.089	0.090	0.084	0.083	0.088	0.092	0.089	0.101	0.101	0.086
TP(5)	0.097	0.098	0.092	0.090	0.093	0.096	0.094	0.096	0.096	0.090
TP(6)	0.099	0.097	0.091	0.090	0.092	0.096	0.093	0.095	0.095	0.090
TP(7)	0.103	0.101	0.095	0.093	0.096	0.099	0.096	0.098	0.098	0.093
TP(8)	0.103	0.101	0.095	0.093	0.096	0.099	0.096	0.098	0.098	0.092
TP(9)	0.103	0.101	0.095	0.093	0.096	0.099	0.096	0.098	0.098	0.092
TP(10)	0.100	0.098	0.093	0.091	0.094	0.097	0.094	0.096	0.096	0.091

Table 5.4: Evaluation using different threshold parameters.

In this experiment we vary the parameters used by TP and TW between 1 and 10 to find the values that give the best F-measure. Based on the results obtained in this experiment the best, parameter for TP is 7 and the best parameter for TW is 1. This threshold parameterization is used in all experiments reported in the following sections.

5.4.3 Evaluation with Different Setups

In this Section we aim to verify the efficiency of the main approach of LX-ListQuestion. We obtain evaluation results using a different setup for each Run, as we explain below.

Experimental Setup: The question dataset used was presented in Table 5.1. The Documents Corpus setup is the same used in the previous experiment (10 retrieved documents). For this experiment we use the following four system setups:

- Run_1: The final list of answers is composed by all elements found on the sentences classified as High Relevance (Premium List). No threshold is applied.
- Run_2: The final list of answers is composed by all elements found on sentences classified as High or Medium Relevance (Premium and Work List). No threshold is applied.
- Run_3: The final list of answers is composed by all elements found on sentences classified as High or Medium Relevance (Premium and Work List). The TW threshold is applied to both lists.

5. EVALUATION

- Run_4: The final list of answers is composed by all elements found on sentences classified as High or Medium Relevance (Premium and Work List) and two thresholds are applied (TP and TW). That is, the full LX-ListQuestion System as described in Section 3.5

The results of this experiment are summarized in Table 5.5.

Experiments	Reference Answer List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
Run_1	340	37	1225	0.108	0.030	0.047
Run_2		72	2369	0.211	0.030	0.053
Run_3		20	146	0.058	0.136	0.082
Run_4		41	460	0.120	0.089	0.102

Table 5.5: Evaluation of the LX-ListQuestion system

Run_1 is our start point to demonstrate how our approach works. In this experiment we verify how many correct answers appear in our Premium List (composed by the candidates extracted from the sentences classified as High Relevance). We found 37 correct answers among 1225 candidates.

In Run_2 we seek to increase recall by appending the Premium List and Work List (composed by the candidates extracted from the sentences classified as High Relevance or Medium Relevance). The system achieves 0.21 recall in Run_2, against 0.10 in Run_1. This shows that the Premium List and Work List together have twice the number of correct answers (The Premium List has 37, while the Premium List and Work List together have 72). However, the precision is low, 0.03 in both Runs.

In Run_3 we seek to filter out the candidate answers to the fullest by applying a threshold, specifically the Threshold Work List (TW_1) - as we presented in Section 3.5.3.2. The goal of this run is to increase the Precision. The precision score increased from 0.03 (Run_2) to 0.13 (Run_3), with a corresponding increase in F-measure of 0.05 to 0.08. Unfortunately, recall decreased. In this run we learned that filters are needed, however, a single filter is very rigid and discard too many correct answers.

Our goal in Run_4 is to increase F-Measure by applying the TP_7 and TW_1 thresholds to the selected candidates in Premium and Work List. As a result, the system doubled the number of correct answers from 20 to 41, while precision only slightly decreased from 0.136

5.5 Comparing LX-ListQuestion and other QA Systems

to 0.089. More importantly, this run leads to an overall F-measure increase from 0.082 to 0.102. Accordingly, the setup from Run_4 will be used for the subsequent experiments.

In this Section, we have shown that using two lists, the Premium and Work lists, ensures better recall. Applying a filter to the candidates is a common strategy to improve precision, but it tends to decrease recall in a way that it does not offset the gains, leading to a worse F-measure. We have also show that this issue can be addressed by using two separate thresholds: a more relaxed threshold which is applied to the Premium List and a more stringent threshold which is applied to the Work List, leading to a better F-measure.

5.5 Comparing LX-ListQuestion and other QA Systems

Comparing LX-ListQuestion with other QA systems is crucial to providing us with an assessment of how LX-ListQuestion is positioned relative to the state-of-the-art. In this Section we compare the results of LX-ListQuestion with four other QA systems: (i) RapPortagico, which runs for Portuguese and uses the same question dataset; (ii) XisQuê, a Web-based QA system, also for Portuguese; (iii) START, a state-of-the-art QA system for English; and (iv) WolframAlpha, a well-known and widely used knowledge engine that can be used as a QA system for English.

The evaluation has two components: the quantitative evaluation of answers and the question coverage evaluation. The quantitative analysis uses precision, recall and F-measure as metrics. These metrics are the most commonly used for evaluating List Questions. As such, these metrics do not accurately reflect how effective the systems are in providing correct answers to the maximum number of questions. For that, we use the question coverage, which determine the number of questions that receive at least one correct answer. This is another dimension under which QA systems can be evaluated, that better indicates the usefulness of the system to the user. This dimension was manually evaluated for all systems.

These experiments were run in the period from November to December, 2014. As such, their replicability cannot be guaranteed since these systems are either Web-based or use a knowledge base that may have been changed.

5. EVALUATION

5.5.1 LX-ListQuestion versus RapPortagico

In this Section we compare the results of LX-ListQuestion and RapPortagico. RapPortagico (Rodrigues and Oliveira, 2012) was the best system of Páigico Competition¹. Table 5.6 shows the differences between the design features of both systems.

	RapPortagico	LX-ListQuestion
Corpus Pre-indexing	Yes. The system pre-indexes the corpus using Noun Phrases.	No
Corpus Source	Off-line Wikipedia documents	Web
Search Engine	Lucene (indexed to documents stored into local files)	Google
Type of answers	List of Wikipedia pages	List of Answers

Table 5.6: Comparing QA systems - RapPortagico and LX-ListQuestion

RapPortagico pre-indexes the documents using noun phrases that occur in the sentences in the corpus while LX-ListQuestion does not use any pre-indexing of documents. RapPortagico uses the off-line Wikipedia as the source of information, while LX-ListQuestion uses the Web to find the answers. The supporting search engines are also different. RapPortagico uses Lucene to find documents indexed in local files and LX-ListQuestion uses Google to retrieve web pages in runtime. Both systems are also different in the type of answers. RapPortagico returns a List of Wikipedia pages and LX-ListQuestion returns a list of answers.

Basically, RapPortagico is an off-line system while LX-ListQuestion is an on-line system. Despite the differences between the two systems, we chose to use RapPortagico as comparison system because the system works for Portuguese and can be directly compared to LX-ListQuestion since both use the same question dataset from the Páigico Competition. The authors kindly provided the output of RapPortagico for data comparison.

Experiments	Reference Answer List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
LX-ListQuestion	340	41	460	0.120	0.089	0.102
RapPortagico		32	327	0.097	0.100	0.098

Table 5.7: Evaluation of QA systems - RapPortagico and LX-ListQuestion

¹ Páigico Competition allowed automatic systems and humans to participate in the competition.

5.5 Comparing LX-ListQuestion and other QA Systems

Experimental Setup: In this experiment we use the same results from Section 5.4.3 and compare the results with the output provided by the authors of RapPortagico.

ID	Correct Answers	
	LX-ListQuestion	RapPortagico
Pagico_053	Parque Estadual Ilha Grande Parque Nacional Tijuca	Floresta Tijuca Parque Nacional Tijuca
Pagico_054	Nossa Senhora Rosario Sao Benedito	—
Pagico_058	Italia	—
Pagico_062	Ericeira Arrifana Praia Vale Homens São João Estoril	— — — —
Pagico_085	—	—
Pagico_088	Obidos	—
Pagico_091	Africa Sul	—
Pagico_092	Damao	Calecute Goa
Pagico_094	Parque Nacional Gorongosa Parque Nacional Limpopo	Parque Nacional Gorongosa —
Pagico_100	Arquipelago Bazaruto Ilha Bazaruto Ilha Santa Carolina Ilha Ibo Ilha Moçambique	Arquipelago Bazaruto Arquipelago Primeiras Segundas Ilha de Santa Carolina Matemo Quirimbas
Pagico_112	Luanda Namibe Luena Ondjdiva Sumbe Uige	Luanda Namibe — — — —
Pagico_140	Salvador Recife São Paulo	Mindelo Cabo Verde — —
Pagico_004	Guilhermina Suggia	Guilhermina Suggia
Pagico_059	—	Leandro Barbosa
Pagico_063	—	Tiago Splitter
Pagico_086	—	—
Pagico_097	Baltasar Lopes Eugenio Tavares	— —
Pagico_104	Celso Magalhaes Lucia Gallet Luis Camara Cascudo Silvio Romero — — —	Atico Vilas-Boas Mota Emilia Biancardi Marco Haurelio Paixao Cortes Raul Lody Saul Alves Martins Vicente Salles
Pagico_106	Afonso Albuquerque Joao Castro D. Luis Francisco Almeida Rei D. Manuel	Constanti Braganca — — — —
Pagico_108	Rolando Nani Varela	—
Pagico_109	—	—
Pagico_111	—	—
Pagico_116	—	Luis Camoes
Pagico_118	Jose Craveirinha	—
Pagico_124	—	—
Pagico_128	— — — — — —	Deolinda Carmo Salvado Conceicao Jose Costa Nunes Jose Rodrigues Santos Jose Silveira Machado Maria Ondina Braga Venceslau Morais
Pagico_132	—	Malangatana
Pagico_133	—	Jose Silva Santana Carlos
Pagico_149	—	—
Pagico_153	—	—

Table 5.8: Comparing answers - LX-ListQuestion and RapPortagico

Table 5.7 shows the results of comparing the two systems. The evaluation was performed for the same set of questions for both systems. As we see, LX-ListQuestion obtained higher

5. EVALUATION

recall than RapPortagico, 0.120 and 0.089 respectively. However, it has lower precision since it returned more candidates than the other system. When comparing F-measure, LX-ListQuestion achieved slightly better results, obtaining 0.102 against 0.095 for RapPortagico.

These scores are too close to allow us to claim a clear superiority of LX-ListQuestion over RapPortagico. We thus turn towards the question coverage evaluation that received at least one correct answer as a way of assessing a different dimension of evaluation.

From the 30 questions in the dataset, LX-ListQuestion provided at least one correct answer to 17 of them, against 14 of RapPortagico. This low rate of effectiveness is due to the fact that the Pagico questions were designed to be non-trivial, requiring a greater effort to answer this type of questions.

The question coverage evaluation (Table 5.8) also allowed us to uncover an interesting behavior of these systems. For 7 questions answered by LX-ListQuestion, RapPortagico provided no answer. Conversely, for 5 questions answered by RapPortagico, LX-ListQuestion provided no answer. In addition, we note that when a question is answered by both systems, the answers given by each system tend to be different.

This result points towards a certain degree of complementarity between both systems. For instance, a system combining the output of LX-ListQuestion and RapPortagico would leave only 8 questions out of 13 unanswered.

5.5.2 LX-ListQuestion versus XisQuê

In this Section we compare the results of LX-ListQuestion and XisQuê. The design features of both systems are to a certain extent similar as show in Table 5.9.

	XisQuê	LX-ListQuestion
Corpus Source	Web	Web
Language	Portuguese	Portuguese
Search Engine	Google	Google
Type of questions	Factoid Questions	List Questions
Type of answers	Answer and Snippet	List of Answers

Table 5.9: Comparing QA systems - XisQuê and LX-ListQuestion

Both systems are Web-base QA systems and use Web as the source of answers, and Google as supporting search engine. In addition, the two systems work for Portuguese.

5.5 Comparing LX-ListQuestion and other QA Systems

What differs between the systems is that the XisQue answers Factoid Questions and LX-ListQuestion answers List Questions. We choose to compare LX-ListQuestion with XisQuê since XisQuê is available online, which means that we can easily perform the experiment. Figure 5.6 shows an example of display of XisQuê System.



The screenshot displays the XisQuê web interface. At the top left is the logo "XisQuê" in orange, followed by the text "Question answering for the Portuguese Web". To the right, there is a text input field labeled "Enter a question:" containing the text "Praias de Portugal boas para a prática de surf". Below the input field are two buttons: "Ask" and "Clear".

Below the input field, the question is displayed: "Question: Praias de Portugal boas para a prática de surf".

Five answers are listed below the question, each with a source document link:

- Answer #1:** Onda para surfistas de todos os níveis. Quando o vento está de noroeste, a praia do Guincho torna-se um dos locais mais ventosos de Portugal, com boas condições para a prática de windsurf e kitesurf. [source document](#)
- Answer #2:** Historial O Guincho é uma das praias onde se pratica surf em portugal desde os anos 60. [source document](#)
- Answer #3:** Portugal está também representado na lista da Condé Nast pelas praias de Peniche, descritas como "um paraíso do surf com maiúsculas" para desfrutar do Verão. "Se é a primeira vez que sobe a uma prancha, as escolas Peniche Surf Camp ensinam-lhe o básico e numa semana estará a 'surfar' com toda a naturalidade", frisa a revista. [source document](#)
- Answer #4:** Clique AQUI para aceder à compilação das melhores praias para a prática de surf da Condé Nast (em espanhol). [source document](#)
- Answer #5:** Boas Notícias - Praias portuguesas entre "paraísos" europeus do surf [source document](#)

Figure 5.6: Display of XisQuê system operation.

Experiment Setup: The evaluation of XisQuê system was done manually by the author of this dissertation. Due to the effort this task requires, only 10 questions (randomly selected from the Question dataset used in previous experiment) were used for this evaluation. For this experiment we used the same results from Section 5.4.3 and compared them with the output provided by the XisQuê Web-system. As XisQuê returns an answer and a snippet, in this assessment, even if the system does not return the correct answer, we consider that the system answered correctly if the answer appears in the snippet. The full set of answers given by XisQuê, with screenshots, may be found in Appendix C.

5. EVALUATION

Table 5.10 shows the evaluation of XisQuê and LX-ListQuestion. Despite both systems having many common design features, the fact that LX-ListQuestion is specifically design to answer List Question allows it to perform better than XisQuê, both in terms of recall and precision, achieving an F-measure of 0.135 against 0.070 of XisQuê.

Experiments	Reference Answer List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
LX-ListQuestion	131	21	179	0.160	0.117	0.135
XisQuê		6	40	0.045	0.150	0.070

Table 5.10: Evaluation of QA systems - XisQuê and LX-ListQuestion

ID	Question	Correct Answers	
		LX-ListQuestion	XisQuê
Pagico_004	Female cellists of portuguese language.	Guilhermina Suggia	—
Pagico_054	Rio de Janeiro churches build by brotherhoods or black fraternities.	Nossa Senhora Rosario São Benedito	—
Pagico_062	Good portuguese beaches for surfing.	Ericeira Arrifana Praia Vale Homens São João Estoril	Guincho Peniche — —
Pagico_086	Brazilian samba songwriters.	—	Dolores Duran
Pagico_088	Portuguese cities that have medieval festivals.	Obidos	—
Pagico_100	Mozambique islands.	Arquipelago Barazuto Ilha Barazuto Ilha Santa Carolina Ilha Ibo Ilha Moçambique	— — — — —
Pagico_109	Applicants for any presidencial elections in Guinea-Bissau.	—	Kumba Yalá
Pagico_112	The capitals of Angola's provinces.	Luena Luanda Namibe Ondjiva Sumbe Uige	Luanda — — — — —
Pagico_133	Petro de Luanda's football players.	—	—
Pagico_140	Lusophone cities known for their carnival celebrations.	Salvador Recife São Paulo	Olinda — —

Table 5.11: Comparing answers - XisQuê and LX-ListQuestion

Regarding the question coverage that we evaluate questions that received at least one correct answer, presented in Table 5.11, we find that a large majority of correct answers

given by XisQuê are different from those given by LX-ListQuestion. Namely, in 4 out of 5 questions to which XisQuê provides a correct answer, that answer is not present in the list of answers given by LX-ListQuestion. In addition, for 2 of the questions (PAGICO_086 and PAGICO_109), XisQuê provides at least one correct answer while LX-ListQuestion gives none. Like with RapPortagico, this suggests that these approaches are complementary.

5.5.3 LX-ListQuestion versus START

START (SynTactic Analysis using Reversible Transformations) is a Web-based question answering system developed by MIT Computer Science and Artificial Intelligence Laboratory (Katz, 1997). Currently, the system can answer millions of English questions about places (e.g., cities, countries, lakes, coordinates, weather, maps, demographics, political and economic systems), movies (e.g., titles, actors, directors), people (e.g., birth dates, biographies), dictionary definitions and others. Figure 5.7 shows an example of display of START QA System.

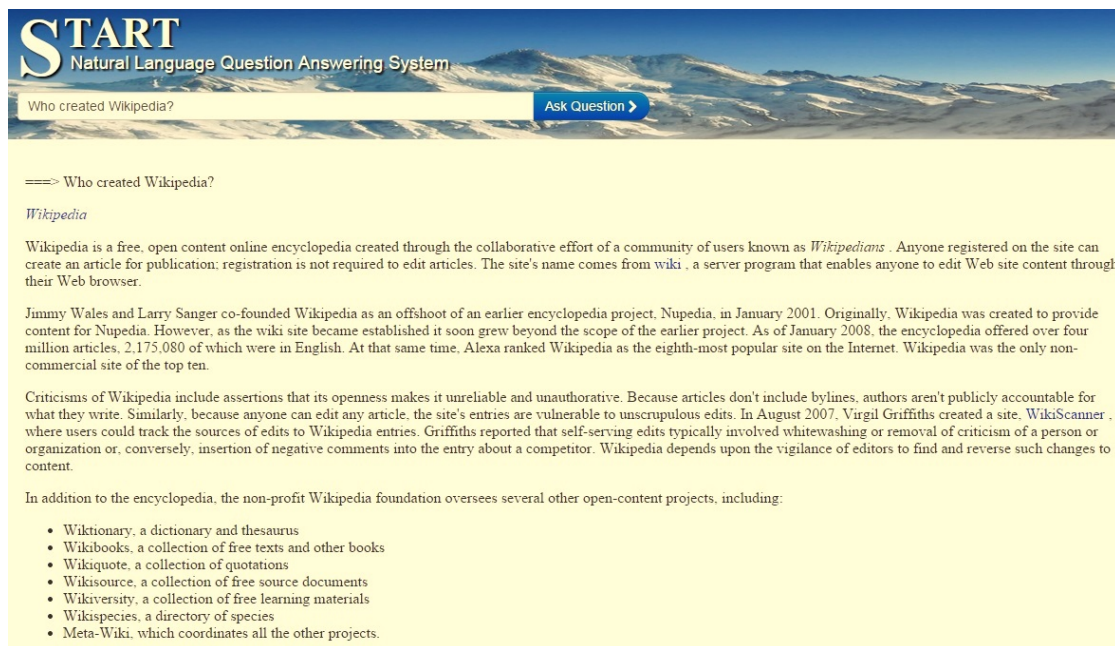


Figure 5.7: Display of START system operation.

The system uses natural language annotation to connect information seekers to multiple information sources. This method allows the system to handle all variety of media, includ-

5. EVALUATION

ing text, diagrams, images, video and audio clips, data sets, Web pages, and others. The NLP component of START consists of two modules that share the same grammar: (i) the understanding module analyzes English text and produces a knowledge base that encodes information found in the text; (ii) the generating module, given an appropriate segment of the knowledge base, produces English sentences. These two modules, used in conjunction with the natural language technique, allow the system to access all stored information.

Table 5.12 compare design features of START and LX-ListQuestion. When comparing START and the LX-ListQuestion, one must take into account that START has been under development over 20 years, while LX-ListQuestion was developed in the scope of this dissertation. The differences between the two systems goes further than the target language. LX-ListQuestion processes the answers in run-time and uses Web-pages as a source of information while START has multiple resources annotated with all information necessary to answer various questions types. We opted for comparing LX-ListQuestion with START since START is a state-of-the-art Web-based QA system and it is available online, which means that we can easily perform the experiment.

	START	LX-ListQuestion
Corpus Source	Multiple Resources	Web
Language	English	Portuguese
Search Engine	N.A.	Google
Type of questions	Factoid Questions Definition Question List Question	List Questions
Type of answers	Answer Summary Images Maps Videos Audio Web-pages etc	List of Answers

Table 5.12: Comparing QA systems - START and LX-ListQuestion

Experiment Setup: The question dataset used was presented in Table 5.2. Similarly to the previous experiment, the evaluation of START system was done manually by the author of this dissertation. As START returns an answer or a short summary of the information, in this assessment, even if the system does not return the correct answer, we consider that the

5.5 Comparing LX-ListQuestion and other QA Systems

system answered correctly if the answer appears in the summary. The full set of answers given by START, with screenshots, may be found in Appendix D.

Experiments	Reference Answer List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
LX-ListQuestion	68	23	171	0.338	0.134	0.192
START		7	18	0.102	0.368	0.160

Table 5.13: Evaluation of QA systems - LX-ListQuestion and START

ID	Question	Correct Answers	
		LX-ListQuestion	START
QALD_010	In which country does the Nile start?	Etiópia —	Ethiopia Rwanda
QALD_028	Give me all communist countries.	Coreia do Norte China Cuba	— — —
QALD_032	Which countries adopted the Euro?	Grecia Eslováquia Estonia Chipre Letonia Lituania Malta	— — — — — — —
QALD_036	Through which countries does the Yenisei river flow?	Russia Mongólia	Russia —
QALD_062	Who created Wikipedia?	Jimmy Wales —	Jimmy Wales Larry Sanger
QALD_074	Which capitals in Europe were host cities of the summer Olympic Games?	Berlim —	—
QALD_114	Give me all members of Prodigy.	Keith Charles Flint	—
QALD_141	Who founded Intel?	Robert Noyce Gordon Moore	— —
QALD_155	Which Greek goddesses dwelt on Mount Olympus?	Afrodite Hera	— —
QALD_176	List the children of Margaret Thatcher.	Carol Thatcher Mark Thatcher	Carol Thatcher Mark Thatcher

Table 5.14: Comparing answers - LX-ListQuestion and START

Table 5.13 shows the evaluation comparing the results of LX-ListQuestion and START. The fact that START uses pre-annotated resources might lead us to expect that START has an advantage over LX-ListQuestion. However, we note that LX-ListQuestion achieved a better F-measure. START has a higher precision score, which is to be expected since START

5. EVALUATION

opts for not providing any answer when it is unable to find any annotation pertaining to the question within its multiple resources. On the otherhand this choice means that START has a much lower recall than LX-ListQuestion, which offsets the gain in precision.

Table 5.14 shows the results of question coverage evaluation. LX-ListQuestion has a better performance giving at least one correct answer to all 10 questions, while START leaves 6 questions unanswered. We also note that, with the exception of two cases (“Rwanda” in QALD_010 and “Larry Sanger” in QALD_062), all the answers provided by START were also found by LX-ListQuestion. Despite START being under development for several years and using multiple annotated resources, this experiment leads us to believe that using the Web as an information source can provide results that are more useful to the user.

5.5.4 LX-ListQuestion versus WolframAlpha

The screenshot displays the WolframAlpha web interface. At the top, the WolframAlpha logo is visible, with the tagline 'computational knowledge engine'. Below the logo is a search bar containing the query 'List the children of Margaret Thatcher.' To the right of the search bar are icons for saving, sharing, and a star. Below the search bar are icons for different input types (text, image, audio, video) and links to 'Examples' and 'Random'. A yellow box below the search bar indicates the system's interpretation: 'Using closest Wolfram|Alpha interpretation: the children of Margaret Thatcher'. The main results area is titled 'Input interpretation:' and shows the query broken down into 'Margaret Thatcher' and 'children'. Below this, the 'Result' is displayed as 'Carol Thatcher | Mark Thatcher'. Further down, the 'Familial relationships:' section is shown, with a 'Show full dates' button. This section includes 'Parents: Alfred Roberts | Beatrice Roberts', 'Sibling: Muriel Roberts', 'Spouse: Denis Thatcher (1951–2003)', and 'Children: Carol Thatcher | Mark Thatcher'.

Figure 5.8: Display of WolframAlpha system operation.

5.5 Comparing LX-ListQuestion and other QA Systems

WolframAlpha is a computational knowledge engine with more than 10 trillion of curate data from primary sources which are continuously updated. The system uses curated data from human experts to compute on the fly a specific answer. WolframAlpha is capable of answering questions in a large variety of fields like mathematics, history, physics, chemistry, medicine, engineering, geography, computational sciences, art, finances, places, people, organizations, music and others¹. The system accepts free-form input. As such, it is possible to use WolframAlpha as a Question Answering system. Figure 5.8 shows an example of display of WolframAlpha.

Table 5.15 shows the comparison of the design features of WolframAlpha and LX-ListQuestion. WolframAlpha and LX-ListQuestion have a few common design features. Strictly speaking, WolframAlpha goes beyond QA, providing alongside the answer much associated information from its knowledge base (images, graphics, maps, etc). Conversely, LX-ListQuestion retrieves its answers from the Web and presents them without any other associated information. We choose to compare these two systems since WolframAlpha is well-known and widely used and is available online allowing us to easily perform the experiment. The full set of answers given by WolframAlpha, with screenshots, may be found in Appendix E.

	WolframAlpha	LX-ListQuestion
Corpus Source	Data Collection	Web
Language	English	Portuguese
Search Engine	N.A.	Google
Type of questions	Factoid Questions Definition Question List Question	List Questions
Type of answers	Answer Summary Images Maps Graphics Tables Sounds	List of Answers

Table 5.15: Comparing QA systems - WolframAlpha and LX-ListQuestion

¹See the complete list of topics in <http://www.wolframalpha.com/examples/> - last access on December, 1 2014.

5. EVALUATION

Experiments	Reference Answer List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
LX-ListQuestion	68	23	171	0.338	0.134	0.192
WolframAlpha		21	39	0.308	0.538	0.392

Table 5.16: Evaluation of QA systems - LX-ListQuestion and WolframAlpha

ID	Question	Correct Answers	
		LX-ListQuestion	WolframAlpha
QALD_010	In which country does the Nile start?	Etiópia	—
QALD_028	Give me all communist countries.	Coreia do Norte China Cuba	— — —
QALD_032	Which countries adopted the Euro?	Grecia Eslováquia Estonia Chipre Letonia Lituania Malta — — — — — — — — —	Vatican City Germany France Italy Spain Netherlands Belgium Austria Greece Finland Ireland Portugal Slovakia Slovenia Luxemburg Cypros Estonia
QALD_036	Through which countries does the Yenisei river flow?	Russia Mongolia	Russia Mongolia
QALD_062	Who created Wikipedia?	Jimmy Wales	—
QALD_074	Which capitals in Europe were host cities of the summer Olympic Games?	Berlim —	—
QALD_114	Give me all members of Prodigy.	Keith Charles Flint	—
QALD_141	Who founded Intel?	Robert Noyce Gordon Moore	— —
QALD_155	Which Greek goddesses dwelt on Mount Olympus?	Afrodite Hera	— —
QALD_176	List the children of Margaret Thatcher.	Carol Thatcher Mark Thatcher	Carol Thatcher Mark Thatcher

Table 5.17: Comparing answers - LX-ListQuestion and WolframAlpha

Table 5.16 shows the evaluation results from comparing the results of LX-ListQuestion and WolframAlpha. The question dataset is the same used in the previous Section to compare LX-ListQuestion and START system. LX-ListQuestion has higher recall than WolframAlpha, 0.338 against 0.308. However, WolframAlpha, due to its manually curated knowledge base, achieves much higher precision, 0.538 against 0.134. This difference in precision leads to an F-measure score of 0.392 for WolframAlpha against 0.192 for LX-ListQuestion.

As in the previous Sections, the question coverage evaluation assesses a different dimension of evaluation and shows a different perspective. Table 5.17 shows the question coverage evaluation. LX-ListQuestion gives at least one correct answer to all 10 questions, while WolframAlpha only provides correct answers to 3 questions. In fact, WolframAlpha is the system that answered the fewest questions from all the systems that were evaluated. Note that the apparently competitive recall score of WolframAlpha comes mostly from QALD_032, which contributes with 17 correct answers.

This stresses the importance of question coverage, since precision and recall scores alone hide much of the actual ability of the system to answer questions. Again, these results show that a system that retrieves answers from the Web can be more useful to the user than a system which uses a curated knowledge base, since a resource is hard to keep up to date with current information.

5.6 Temporal List Questions: Evaluation

In this section we evaluate the Temporal Processing module of LX-ListQuestion system that was presented in Chapter 4. We start with an experiment that aims to test our approach for different temporal restrictors over the same base question.

Following this, we evaluate the LX-ListQuestion with Temporal Processing module using a subset of questions from Páxico Competition randomly selected from those with temporal restrictions.

The results are compared with another QA System using the quantitative evaluation of answers and the question coverage evaluation. Since the question dataset of the Páxico Competition was built to find the answers in Wikipedia and LX-ListQuestion is a Web-based QA system that uses all the Web as information source, we evaluate the system over another question dataset.

5. EVALUATION

This dataset is composed of five questions with temporal restriction where both the question and the answers were given by humans. The questions were retrieved from YAHOO!Answers¹. Finally, we compare the results of these question dataset with another Web-based QA System.

5.6.1 Evaluation for Different Temporal Restrictors

In order to evaluate the Temporal Processing Module of our system we perform an assessment exercise for different temporal restrictors. We decided to use the same base question and measure the performance of LX-ListQuestion when different temporal restrictors are added to the question.

The base question is the same used to explain our approach in the Chapter 4, “Give all books from Danielle Steel”. We especially chose this question since the author, Danielle Steel, has many publications spanning several years, which allows us to apply a large variety of temporal restrictors for the same base question.

Experimental setup: Our experiment is the Temporal Processing Module embedded into the LX-ListQuestion system to assess and evaluate the results of the same question using different temporal restrictors. The results are presented in Table 5.18.

Question	Reference Answer List	#Correct Answers	#Answers Retrieved	Recall	Precision	F-Measure
Quais são os livros da Danielle Steel ?	79	76	259	0.962	0.293	0.449
Quais são os livros da Danielle Steel antes de 1990 ?	27	26	73	0.963	0.356	0.520
Quais são os livros da Danielle Steel nos últimos 15 anos ?	31	29	54	0.935	0.537	0.682
Quais são os livros da Danielle Steel na década de 90 ?	26	23	36	0.884	0.638	0.741
Quais são os livros da Danielle Steel entre 1985 e 1990 ?	9	9	17	1.000	0.529	0.692
Quais são os livros da Danielle Steel em 1981, 1991 e 2001 .	9	9	14	1.000	0.529	0.692
TOTAL	102	96	194	0.941	0.494	0.648

Table 5.18: Evaluation for different temporal restrictors.

Note that the first line of Table 5.18 shows the results for the base question (i.e. with no temporal restriction). When the system answers the question without any temporal restriction, it achieves a lower precision since lots of incorrect answers are selected in the

¹<https://br.answers.yahoo.com/> - last access on October, 20. 2014.

final list answer. In the following questions we vary the temporal restriction initially using a broader restrictor (“...antes de 1990.”) and then gradually restricting the timespan of questions (“...em 1981, 1991 e 2001.”).

In general, for these questions LX-ListQuestion with the Temporal Processing module achieves a satisfactory performance, and excellent recall score (between 0.884 and 1.000). We observe that the system tends to get a higher precision score when the temporal restriction is narrower, as we can see in the question “Quais são os livros da Danielle Steel em 1981, 1991 e 2001.” which gets a higher precision than “Quais são os livros da Danielle Steel antes de 1990?”. This happens because larger timespans allow for more wrong candidates to be chosen. For instance, the restrictor “...antes de 1990.” (ENG: “...before 1990.”) allows for any number lower than 1990.

5.6.2 Evaluation of Temporal List Question over Questions from Págico

In this Section we evaluate LX-ListQuestion with the Temporal Processing module using five questions from the Págico Competition dataset randomly selected from those with temporal restriction. The questions and the number of correct answers provided in the reference list of answers are presented in Table 5.19.

ID	Question	Reference List of Answer
Pagico_008	Telenovelas brasileiras passadas no tempo da escravidão no Brasil	7
Pagico_034	Viajantes ou exploradores que escreveram sobre o Brasil do século XVI	26
Pagico_050	Jornais que circularam no Rio de Janeiro entre 1910 e 1960.	2
Pagico_068	Bandas brasileiras de punk formadas até 1980 em São Paulo.	9
Pagico_078	Escritoras de língua portuguesa que tenham publicado livros para crianças entre 1850 e 1940	8

Table 5.19: A set of question dataset of Págico Competition.

In order to evaluate the results we compare the same questions with RapPortagico QA system (already presented in Section 5.5.1). In general, both systems struggle to answer questions of Págico Competition. As mentioned earlier, the questions provided by Págico are non-trivial and require a complex processing by the systems. The scores are very similar, with a slightly advantage for LX-ListQuestion of 0.114 in F-Measure against 0.098 of RapPortagico, with no system standing out as the clear winner.

5. EVALUATION

ID	LX-ListQuestion					RapPortagico				
	Correct Answers	Answers Retrieved	Recall	Precision	F-Measure	Correct Answers	Answers Retrieved	Recall	Precision	F-Measure
Pagico_008	0	0	0.000	0.000	0.000	3	6	0.176	0.500	0.260
Pagico_034	1	5	0.125	0.200	0.153	1	22	0.125	0.045	0.066
Pagico_050	0	0	0.000	0.000	0.000	0	21	0.000	0.000	0.000
Pagico_068	4	31	0.444	0.129	0.200	3	24	0.333	0.125	0.181
Pagico_078	0	0	0.000	0.000	0.000	0	18	0.000	0.000	0.000
TOTAL	5	36	0.098	0.138	0.114	7	91	0.137	0.076	0.098

Table 5.20: Evaluation of temporal list questions: LX-ListQuestion versus RapPortagico.

Similarly to what was done in the previous Section, a question coverage evaluation is used to obtain another dimension to the QA evaluation. The question coverage evaluation can indicate the usefulness of the QA system. The question coverage is presented in Table 5.21 and reveals the same behavior found in the previous analysis (presented in Table 5.8). It points to a certain degree of complementary between both systems since, for question Pagico_008, the RapPortagico provides answers and LX-ListQuestion provides no correct answers. Conversely, for question Pagico_034, LX-ListQuestion provided a correct answer and RapPortagico provides none. For question Pagico_068 the answers provide by both systems are different.

ID	Question	Correct Answers	
		LX-ListQuestion	RapPortagico
Pagico_008	Telenovelas brasileiras passadas no tempo da escravidão no Brasil	—	Sangue do Meu Sangue Sangue do Meu Sangue 1969 Sangue do Meu Sangue 1995
Pagico_034	Viajantes ou exploradores que escreveram sobre o Brasil do século XVI	Hans Staden	—
Pagico_050	Jornais que circularam no Rio de Janeiro entre 1910 e 1960.	—	—
Pagico_068	Bandas brasileiras de punk formadas até 1980 em São Paulo.	Condutores Cadaver Ratos do Porão Restos Colera	Lixomania Olho Seco Restos
Pagico_078	Escritoras de língua portuguesa que tenham publicado livros para crianças entre 1850 e 1940	—	—

Table 5.21: Question coverage: LX-ListQuestion versus RapPortagico.

5.6.3 Evaluation of Temporal List Question over Questions from Yahoo! Answers

In this Section we evaluate LX-ListQuestion with the Temporal Processing module using five questions with temporal restriction retrieved from YAHOO! Answers. Note that these

5.6 Temporal List Questions: Evaluation

are actual questions posted by users, a kind of question that LX-ListQuestion aims to answer. The questions and the number of correct answers provided in the reference list are presented in Table 5.22.

ID	Question	Reference List of Answer
TP_001	Quais eram os partidos politicos existentes antes de 1964?	7
TP_002	Quem ganhou o premio nobel entre 1900 a 1920?	100
TP_003	Quais foram as novelas brasileiras dos últimos 5 anos?	10
TP_004	Que países boicotaram os Jogos Olímpicos de 1980?	65
TP_005	Quais são as bandas brasileiras de rock dos anos 80?	45

Table 5.22: Temporal questions given by Yahoo!Answers.

The results of LX-ListQuestion and XisQuê are presented in Table 5.23. LX-ListQuestion achieves better recall that improves F-measure score from 0.243 against 0.054 of XisQuê. As we mentioned earlier, in Section 5.5.2, LX-ListQuestion stand out with better results since that it have special module to answer Temporal List Questions.

ID	LX-ListQuestion					XisQuê				
	Correct Answers	Answers Retrieved	Recall	Precision	F-Measure	Correct Answers	Answers Retrieved	Recall	Precision	F-Measure
TP_001	7	46	1.000	0.152	0.264	2	5	0.285	0.400	0.333
TP_002	26	108	0.260	0.240	0.250	0	0	0.000	0.000	0.000
TP_003	2	7	0.200	0.285	0.235	0	5	0.000	0.000	0.000
TP_004	9	24	0.138	0.375	0.202	0	1	0.000	0.000	0.000
TP_005	8	16	0.177	0.500	0.262	5	17	0.111	0.294	0.161
TOTAL	52	201	0.229	0.258	0.243	7	28	0.038	0.250	0.054

Table 5.23: Evaluation of temporal list questions: LX-ListQuestion versus XisQuê.

In the question coverage evaluation presented in Table 5.24 we can see the superiority of LX-ListQuestion in terms of providing correct answers to all number of questions. The question coverage evaluation, in this case, supports the results obtained in the quantitative evaluation. Although, its important stand out that even the lower recall, for the question TP_005, XisQuê provides some distinct answers from LX-ListQuestion. This suggest that the systems may have a complementary approaches.

5. EVALUATION

ID	Question	Correct Answers	
		LX-ListQuestion	XisQuê
TP_001	Quais eram os partidos políticos existentes antes de 1964?	Arena Movimento Democratico Brasileiro Partido Trabalhista Partido Socialista Uniao Democratica Nacional Partido Comunista Partido Trabalhista Brasileiro	Arena Movimento Democratico Brasileiro
TP_002	Quem ganhou o premio nobel entre 1900 a 1920?	Paul Sabatier Jacobus Henricus van Wilhelm Conrad Rontgen (...21 answers omitted...) Sir William Bragg Adolf Von Baeyer	
TP_003	Quais foram as novelas brasileiras dos últimos 5 anos?	Passione Fina Estampa	
TP_004	Que países boicotaram os Jogos Olímpicos de 1980?	Japao Estados Unidos (...5 answers omitted...) Hong Kong Chile	
TP_005	Quais são as bandas brasileiras de rock dos anos 80?	Titas Paralamas sucesso Blitz Joao Penca Cazuza Barao Vermelho Eduardo Dusek Angra	Titas Paralamas sucesso Legião Urbana Kid Abelha Lobão

Table 5.24: Question coverage: LX-ListQuestion versus XisQuê.

5.7 Summary

This Chapter detailed the evaluation process carried out to assess the performance of the approaches presented in Chapters 3 and 4. We divided the evaluation in three parts.

In the first part, we evaluated separately each parameter of LX-ListQuestion to find the best setup for the configuration of our system. We tested different scenarios, with corpora of different size, and opted for setting the number of retrieved documents at 10 (default number in the state-of-the-art) since the other scenarios do not perform any better. Then we verified how many correct answers our system finds using different threshold boundaries. Based on the experiment presented, the best parameter for TP is 7 and the best parameter for TW is 1 because these values achieve the best F-Measure score.

Following this, we assessed the efficiency of our approach by comparing the results using four different setups. We concluded that using two lists, the Premium and Work lists, ensures better recall. We demonstrated that using two separate thresholds, a more relaxed threshold which is applied to the Premium List and a more stringent threshold which is applied to the

Work List, leads to a better F-measure.

In the second part of this Chapter, we compared LX-ListQuestion with other QA systems. This comparison is important to assess how LX-ListQuestion is positioned with respect to the state-of-the-art. LX-ListQuestion was compared with four other QA systems: (i) RapPortagico, an off-line List and Factoid QA system for Portuguese; (ii) XisQuê, a Web-based Single Factoid QA system, also for Portuguese; (iii) START, a state-of-the-art List and Factoid QA system for English; and (iv) WolframAlpha, a well-known and widely used knowledge engine that can be used as a QA system for English.

For the comparison, we used two analysis: (a) a quantitative evaluation of answers providing recall, precision and F-measure using the same dataset for each system; and (b) a question coverage evaluation that better indicates the usefulness of the system to the user, ensuring another dimension of evaluation. This comparison brings interesting results. Pointing to a degree of complementarity of LX-ListQuestion when comparing the question coverage of RapPortagico and XisQuê.

Regarding the QA systems for English, our analysis shows that, even though START uses pre-annotated resources which might lead us to expect that START had an advantage, LX-ListQuestion achieved a better F-measure. The question coverage evaluation consolidates the results obtained from the quantitative evaluation.

The quantitative results of the comparison of LX-ListQuestion and WolframAlpha indicates that WolframAlpha achieves higher precision since it uses a manually curated knowledge base as the source of information. The question coverage evaluation assesses a different dimension of evaluation and shows a different perspective. WolframAlpha is the system that answered the fewest questions from all the systems that were evaluated and the apparently competitive recall score of WolframAlpha comes mostly from one question, which contributes with 17 correct answers; while LX-ListQuestion gives at least one correct answer to all the questions in the dataset.

The last part of this Chapter refers to the evaluation of the Temporal Processing module of LX-ListQuestion. First, we tested our approach for different temporal restrictors over the same base question. LX-ListQuestion with the Temporal Processing module achieves a satisfactory performance, and excellent recall score (between 0.884 and 1).

Following this, we evaluate the system using the question dataset made available at Págico Competition and compare the results against RapPortagico. Our experiment shows

5. EVALUATION

that both systems struggle to answer questions of Páxico Competition since the questions provided by Páxico are non-trivial and require complex processing by the systems.

Another dataset was built extracting questions with temporal restriction retrieved from YAHOO! Answers. Note that these are actual questions posted by human users. For this dataset, LX-ListQuestion achieves better recall that improves F-measure score to 0.243, against 0.054 of XisQuê. The question coverage evaluation for the comparison between LX-ListQuestion, RapPortagico and XisQuê indicates, the same way as mentioned before, a degree of complementarity.

Final Remarks

This chapter evaluated extensively our approaches to tackling List questions and Temporal List questions. Our approach to answer List questions is a novelty since it combines redundancy and heuristics. The evaluation performed in this chapter showed that our approach achieves better results when comparing with other QA system and improved the state-of-the-art.

Regarding Temporal QA, we addressed this issue by combining the Web-based approach to collect all possible answers to the question with a shallow temporal processing approach to filter the answers based on the temporal restriction. The experiments in this Chapter showed that we achieved competitive results. Our approach to answer Temporal List questions using a Shallow Temporal Processing is the first working system combining Temporal, List and Web-based Question Answering and it is a valuable contribution in this field.

6

Conclusions

This dissertation addressed the task of answering Open-domain List questions whose answers are extracted from multiple documents retrieved from the Web, with special focus on questions that include temporal information. This final chapter is organized as follows: Section 6.1 gives a short overview of what is covered in each Chapter. The main goals and contributions of our work are presented in Section 6.2. Finally, some future directions of research are discussed in Section 6.3.

6.1 Summary

The contents of this dissertation are summarized as follows:

Introduction: Chapter 1 is a general introduction to the area of Question Answering that addressed specific subareas: Open-Domain QA, Web-based QA and Temporal QA. We presented the type of questions more currently studied. Motivation, goals and challenges for the research in this area were also presented.

Related Work: Chapter 2 is a review of the current state-of-the-art. We presented QA systems developed for Portuguese. Some of these systems were developed to participate in Competitions like CLEF, GikiCLEF or Págico. For Portuguese, we highlighted the XisQuê system, a Web-based QA system that answers factoid questions and uses linguistic patterns as the main approach. We also presented QA systems that handle List questions. The main approaches exploit (i) NLP tools and linguistic resources; (ii) the relation between question

6. CONCLUSIONS

and possible answers; and (iii) the semantic content. For List questions we highlighted Rap-Portagico, a QA system developed to answer List questions by searching for answers using an off-line Wikipedia as information source, which was the system with best performance in the Páxico Challenge.

Still in this Chapter 2 Web-based QA systems are presented. There is a great variety in the approaches developed that use Web as a corpus, such as those that explore redundancy, probabilistic models, clustering, etc. Most systems answer only factoid questions. The START system developed by MIT and available since 1993 handles List questions as well.

In the end of Chapter 2, we presented the approaches to answer Temporal questions. Our overview indicates that a predominant approach did not emerge yet since each researcher chose a different path depending on the type of temporal questions. Some works use the Web as a corpus to answer temporal questions. However, these works try to extract information and store it into knowledge bases to enable the QA system to access information more easily, thus needing constant effort to keep the database up to date.

Answering List Question: Chapter 3 covers a major goal of this dissertation, developing an approach for processing List questions and collecting answers spread over multiple documents using the Web as a corpus. Our approach is based on redundancy of information available on the Web combined with heuristics to improve QA performance.

Our approach takes advantage of the sentences being classified according to their relevance to the question. Our approach, termed Bipartite List, is based on building two lists in which each element in the list is associated to its frequency. Afterwards, two empirically determined thresholds (one for each list) are applied to select the potentially relevant answers. Besides this process, we developed three heuristics: (1) Verb-Rule: selects a candidate as an answer if the sentence in which that candidate occurred contains the same verb of the question; (2) Title-Rule: selects a candidate as an answer if it was extracted from texts in which the text title matches (i.e. all keywords are present) the question ; and (3) Sentence Match-Rule: selects candidate as an answer if it was extracted from sentences that match the root question.

We implemented our approach into a fully-fledged Open-domain Web-based QA system for List questions. The system architecture is composed by three main modules: Question Processing, Passage Retrieval and Answer Extraction. The Question Processing module is

responsible for converting a natural language question into a form that subsequent modules are capable of handling. The main sub-tasks are (i) question analysis: responsible for cleaning the questions, i.e. removing question marks, interrogative pronouns and imperative verbs; (ii) extraction of keywords: performed using two different algorithms, namely Nominal Expansion and Verbal Expansion; (iii) transformation of the question into a query; (iv) identification of the semantic category of the expected answer; and (v) identification of the question-focus.

The Passage Retrieval Module is responsible for searching Web pages and saving their full textual content into local files for processing. After the content is retrieved, the system will select relevant sentences. The Answer Extraction Module aims at identifying and extracting relevant answers and presenting them in list form. The candidate answer identification is based on a Named Entity Recognition tool. The candidates are selected if they match the semantic category of the question. The process of building the Final List Answer is based on frequency and heuristics as mentioned before. As discussed, this novel approach allows finding a better balance between precision and recall.

Answering Temporal List Question: Chapter 4 provided the background for Temporal Question Answering and presented the main concepts and challenges. We presented an overview of the state-of-the-art in Temporal QA. We presented the design features of our Temporal Processing Module and the questions that are expected as input. We explained how we solved the time-range issue of the temporal expressions. We also presented our approach to shallow temporal processing to find answers that satisfy the temporal restrictions in the questions. Our approach handles questions with (i) absolute time restriction (e.g. “*Que países boicotaram os Jogos Olímpicos de 1980?* EN: Which countries boycotted the 1980 Olympic Games?”) and (ii) relative time restriction (e.g. “*Quais eram os partidos políticos existentes antes de 1964?* EN: Which political parties existed before 1964?”). For the Temporal Processing, it is very important to have the maximum number of candidates for the system to start the processing. For each candidate, we extracted from the full set of documents every temporal expression that co-occurs in the same sentence with that candidate. If the temporal expression extracted satisfies the temporal restriction in the question, the candidate is accepted into the final list of answers.

Evaluation: In Chapter 5, we evaluated separately each parameter of LX-ListQuestion to find the best setup for its configuration. We compared also LX-ListQuestion with other four QA systems: (i) RapPortagico, an off-line List and Factoid QA system for Portuguese;

6. CONCLUSIONS

(ii) XisQuê, a Factoid Web-based QA system, also for Portuguese; (iii) START, a state-of-the-art QA system for English; and (iv) WolframAlpha, a well-known and widely used knowledge engine that can be used as a QA system for English. For the sake of comparison, the results were analyzed in two ways: (1) The quantitative evaluation of answers provides recall, precision and F-measure, comparing LX-ListQuestion with each one of the other systems, under same question dataset. (2) The question coverage indicates the usefulness of the system to the user by counting the number of questions that the system provides at least one correct answer.

Compared with the systems for Portuguese, RapPortagico and XisQuê, our LX-ListQuestion achieved better results, with 0.102 in F-Measure, against 0.098 of RapPortagico, and 0.070 of XisQuê. The question coverage evaluation points towards a certain degree of complementarity between these systems. We observe that for a set of questions answered by LX-ListQuestion, the other systems provide no answers. Conversely, for some other questions answered by RapPortagico or XisQuê, LX-ListQuestion provided no answer. In addition, we note that when a question is answered by the three systems, the answers given by each system tend to be different.

Regarding the comparison with systems for English, START and WolframAlpha, our LX-ListQuestion achieved better results than START, with 0.192 F-Measure against 0.160, while WolframAlpha performed better than the other two, achieving 0.392 F-Measure. The question coverage evaluation revealed a more detailed scenario. LX-ListQuestion has a better performance giving at least one correct answer to all 10 questions in our experiment, while START leaves 6 questions unanswered and WolframAlpha leaves 7 questions unanswered. In fact, WolframAlpha is the system that answered the fewest questions from all the systems that were evaluated. This stresses the importance of question coverage evaluation, since precision and recall scores alone hide much of the actual ability of the system to answer questions.

In the end of Chapter 5, we evaluated more closely the ability of LX-ListQuestion with respect Temporal questions. We tested our approach for five different temporal restrictors over the same base question, “*Quais são os livros da Danielle Steel? EN: Which are Danielle Steel books?*”. Our approach achieved a positive performance and excellent recall score achieving 0.941 on average.

The evaluation proceed using two different datasets of Temporal List Questions. The first dataset was composed of questions from Págico Competition and the performance were com-

pared against RapPortagico. Our experiment showed that both systems struggle to answer these questions since the questions provided by Páxico are non-trivial and require complex processing by the systems.

Another dataset was built collecting questions with temporal restriction from YAHOO! Answers, which are actual questions posted by human users. The results were compared with those from XisQuê. For this dataset, LX-ListQuestion achieves better recall that improves F-measure with a score of 0.243, against 0.054 of XisQuê. The assessment of question coverage evaluation for the Temporal Processing Module between LX-ListQuestion, RapPortagico and XisQuê indicates, as before, a degree of complementarity.

6.2 Contributions

The present dissertation achieves several goals and makes a number of contributions to the research in the field of Open-domain Web-based List QA and Temporal QA. Here, we review the contributions proposed in Chapter 1.

Developing an approach to deal with List questions: We developed an approach to extract a list of answers spread over multiple documents using the Web as a corpus (Gonçalves and Branco, 2014a). Our approach is based on the redundancy found in the Web combined with heuristics. Our proposed approach consists of three parts:

1. **Candidate retrieval:** The approach selects and classifies sentences according to their relevance to the question using the number of keywords as score. After this classification, the sentences are used to select all candidate answers in two ways: (i) by using a named entity recognizer tool to classify and select the candidates that match the expected semantic type of the question and (ii) by picking candidates that match the focus conveyed by the question.
2. **Bipartite list approach:** Our approach exploits redundancy to find all answers to the List questions, and uses their frequency as a factor to select the correct answer. We build two lists: (i) Premium List, which is composed by candidates extracted from the sentences with high relevance to the question and (ii) Work List, which is composed by candidates extracted from the sentences with weak relevance to the question. In the each list, the candidates that appear repeated are grouped together and their frequency

6. CONCLUSIONS

is calculated. We use two frequency thresholds, one threshold to filter the candidates of Premium List and the other to filter the candidates of Work List. The final list of answers are composed by the candidates filtered by the thresholds.

3. Heuristics: We developed and applied three heuristics based on word occurrence: (i) Verb-Rule, which selects a candidate as an answer if the candidate appears in a sentence with the same verb given by the question; (ii) Title-Rule, which selects as an answer all candidates from documents whose title matches the question; and (iii) Sentence Match-Rule, which selects as answer all candidates extracted from sentences that match the root-question.

Our approach for dealing with List questions is novel since other methods gather the candidates in a single list (or cluster), meaning that their frequency filters are applied to every candidate. Our evaluation has shown that our system achieves better results when compared with other QA system.

Previous approaches use either frequency-based or rule-based filters to select candidates. Our approach combines frequency and heuristics based on rules that provides a novel contribution and shows that a combined approach improves the state-of-the-art.

Developing an approach to deal with Temporal List questions: Our processing method has three main steps:

1. Identifying the temporal expression: The temporal expression is identified using hand-build patterns. We based our patterns in a corpora study in which we identify the most common temporal expressions in the Páxico and QALD question datasets. We presented these patterns in detail in Chapter 4.
2. Transforming temporal expression into a temporal constraint: After the identification of the temporal expression, we have to anchor this temporal expression in a calendar year, defining the temporal boundaries of the temporal expressions. This is an important design feature since the temporal expression can be expressed as an absolute time restriction (e.g. “List films of 2014”) or as a relative temporal reference (e.g. “List films after 1995”). Our approach determines the time-range of these two types of temporal expressions.

3. Shallow temporal processing: For each answer candidate, we extract from the full set of documents every temporal expression that co-occurs in the same sentence with that candidate. The candidate is accepted into the final list of answers if any of the temporal expressions lie within the boundary defined by the temporal constraint.

Temporal List Question Answering is still an under-researched topic. In our review of the state-of-the-art, we found Web-based approaches to temporal questions but which only handle Factoid Temporal questions. To the best of our knowledge, our contribution is the first working approach that concomitantly addresses Temporal, List and Web-based Question Answering.

Implementing our approaches into a fully-fledged Web-based QA system: The evaluation that was performed has shown the validity of the approach we developed. Additionally, further important contribution is the implementation of LX-ListQuestion (Gonçalves and Branco, 2014b) as a freely available online service for answering List questions¹.

The system developed in this dissertation uses Google as a search engine to collect the relevant documents. LX-ListQuestion provides answers in real-time without resorting to previously stored information. To account for different styles of querying, the system allows multiple types of input questions: (i) a syntactically correct interrogative sentence; (ii) an imperative sentence or (iii) a keyword-based query. Being Web-based, LX-ListQuestion is not tied to a fixed pre-processed database. As such, it handles information that may change quickly over time. All these features mean that the LX-ListQuestion is a robust Web-based QA system, making it a valuable contribution to the field of Question Answering.

6.3 Future Research Directions

There are several challenges that can be addressed in the future in order to improve Question Answering, in general, and List Questions and Temporal Questions, in particular.

In our research, we have found that List questions can be even more complex in several ways. Our corpus study of questions from Páigico and QALD brings out a few cases of complex list questions:

¹ Available at <http://lxlistquestion.di.fc.ul.pt>

6. CONCLUSIONS

- Relative questions: Usually, this type of questions starts with a relative clause constituent (In which), and the referent appears explicit in the questions.

QALD_025 - **In which** films directed by Garry Marshall was Julia Roberts starring?

- Multiple Question-focus: This type of question where there is more than one focus: “Jornais” and/or “revistas” and/or “publicações periódicas”

PAGICO_138 - **Jornais, revistas** e outras **publicações periódicas** de Macau.

(EN: **Newspaper, magazines** and other **publications** of Macau.)

- Negative questions: These are interrogative sentences which contain negation in their phrasing:

PAGICO_113 - Ilhas e ilhotas de Cabo Verde que **não** são habitadas.

(EN: Cape Verde islands and islets that are **not** inhabited.)

In our assessment of the state-of-the-art, we did not find any approach that handles these more complex types of List questions. All these complex types of questions need a different treatment to find the correct answer, where more elaborate approaches need to be studied and developed.

In what concerns more specifically List questions, on the basis of our experiments, we learned that only increasing the corpus size (getting a larger number of documents) does not guarantee higher F-Measure on List questions, since retrieving more candidates also brings additional wrong answers. Possible ways to resolve this issue are fairly unexplored. Initial steps towards addressing this issue were given by Yang and Chua (2004a), who proposed a framework that applies categorization at the level of the Web page aiming to extract distinct answers to List questions, attempting to increase recall without sacrificing precision. Since this work was developed, more than 10 years ago, little was done to tackle this problem, which still leaves room for developing new approaches.

The basis of on our experiments in Chapter 5, we noted that the approaches of Rap-Portagico, XisQuê and LX-ListQuestion may reinforce each other. LX-ListQuestion is a Web-based QA system that uses redundancy and heuristics to answer List questions. Rap-Portagico is an off-line QA system that uses Wikipedia to retrieve the answers for List questions. Since it uses a structured resource as its source of information, it can answer with higher precision than LX-ListQuestion. XisQuê is a Web-based QA system that answers factoid questions that selects the most important paragraph of the Web pages and extracts

the answer through the use of hand-built patterns. As such, low frequency precise answers are not necessarily discarded, as it may happen when the frequency thresholds used by LX-ListQuestion are applied.

The idea of combining several QA system for Portuguese has been proposed before in Carvalho *et al.* (2010), where a hypothetical combination of six different QA system for Portuguese was considered. They demonstrate that the hypothetical system would achieve better results since each individual system was good in answering certain type of questions. To support this suggestion, we built Table 6.1 with an overview of the results obtained in Chapter 5. The last row is the hypothetical combination of LX-ListQuestion, RapPortagico and XisQuê. As we can see, a QA system that combines their approaches can achieve better results and improve Recall and F-measure metrics.

Systems	Reference Answers List	Correct Answers	All Answers Retrieved	Recall	Precision	F-Measure
LX-ListQuestion	340	41	460	0.120	0.089	0.102
RapPortagico		32	327	0.097	0.100	0.098
XisQuê		6	40	0.014	0.128	0.026
Combination		72	819	0.211	0.087	0.124

Table 6.1: Results overview

Considering Temporal QA, our approach is focused on temporal expressions related to years and centuries. There are a lot of other temporal expressions. A more robust approach to the identification of other temporal expressions should be designed.

- (1) Examples of other temporal expressions:
 - a. in the Summer of 95.
 - b. in the last hours of 1999/12/31.
 - c. in the morning of September 11, 2001.

Still regarding temporal expressions, the research under vague temporal expressions needs be deepened. Schockaert *et al.* (2006) addresses the problem of answering questions with vague temporal information: the fact that many historical events cannot be accurately captured by an interval with well-defined boundaries. They propose to build a knowledge base automatically by extracting the information from Wikipedia using a simple pattern-based approach. We believe that the same strategy can be easily replicated for Portuguese

6. CONCLUSIONS

and a QA system can take advantage of this to improve the results in the Temporal QA field.

- (2) Examples of vague temporal expression in Portuguese context:
- a. during the Age of Portuguese Discovery.
 - b. before the government of Salazar.
 - c. after the great Lisbon earthquake.

Complex Temporal List questions is an under researched topic. Answering such questions is a non trivial task due to the potential complexity of the questions. In our overview of the state-of-the-art, we did not find any work considering Complex Temporal expressions in List questions. Essentially this field is connected to identifying the temporal relations between events which are explicitly marked by a temporal prepositions (*before, at, on, starts, etc*). The approach proposed by Schilder and Habel (2003) uses a temporal tagger in order to annotate and automatically extract temporal expression and their relations with events. Recently, Costa and Branco (2013) have developed a tool named LX-TimeAnalyzer that extracts temporal information from Portuguese text, aimed at finding the following elements: (i) Temporal expressions, which are expressions that occur in the input text and that refer to dates and times; (ii) Events terms, which are words that refer to events that happen or hold at some point in time; (iii) Temporal relations between these times and events, i.e. the temporal ordering among these entities (*before, after, overlap*), according to the input text. A QA system integrated with a tool like LX-TimeAnalyzer can improve the state-of-the-art in Temporal QA.



Question Dataset

This appendix lists the full set of questions used in our experiments. The English translation of the questions is also shown, as well as the list of correct answers found in the reference list.

A.1 Págico Dataset

Below, you may find the subset of questions from the Págico competition and their correct answers that were used in our experiments.

Pagico_004 Mulheres violoncelistas de língua portuguesa.

Portuguese-language female cellists.

Carmen Monarcha, Denise Emmer, Guilhermina Suggia

Pagico_053 Parques do Rio de Janeiro que têm cachoeiras.

Parks with waterfalls in Rio de Janeiro.

Floresta da Tijuca, Parque Estadual da Ilha Grande, Parque Nacional da Tijuca

Pagico_054 Igrejas do Rio de Janeiro construídas por irmandades ou confrarias de negros.

Churches in Rio de Janeiro built by black religious brotherhoods or fraternities.

Igreja de Nossa Senhora do Rosário e São Benedito Rio de Janeiro, Rio de Peixe Mina

A. QUESTION DATASET

Pagico_058 Países que venceram a Copa do Mundo em uma disputa de pênaltis.

Countries that won the World Cup by penalty shootouts.

Brasil, Itália

Pagico_059 Jogadores de basquete brasileiro que jogam ou jogaram em campeonatos da NBA.

Brazilian basketball players that play or have played in the NBA.

Alex Ribeiro Garcia, Anderson Varejão, Leandro Barbosa, Marcus Vinicius de Souza, Maybyner Rodney Hilário, Paulo Sérgio Prestes, Rafael Paulo de Lara Araújo, Tiago Splitter

Pagico_062 Praias de Portugal boas para a prática de surf.

Good Portuguese beaches for surfing.

Baleal, Cabedelo do Douro, Cortegaça Ovar, Costa da Caparica, Ericeira, Mindelo Vila do Conde, Peniche, Praia Grande Sintra, Praia da Aguçadoura, Praia da Albandeira, Praia da Amoreira, Praia da Areia Branca, Praia da Arrifana, Praia da Barranha, Praia da Consolação, Praia da Foz do Lizandro, Praia da Lagoa, Praia da Lagoa de Albufeira, Praia da Memória Matosinhos, Praia da Nazaré, Praia da Ribeira d Ilhas, Praia da Salgueira, Praia de Carcavelos, Praia de Santo Amaro de Oeiras, Praia de Vale dos Homens, Praia do Amado, Praia do Areal, Praia do Beliche, Praia do CDS, Praia do Cabedelo Viana do Castelo, Praia do Furadouro, Praia do Guincho, Praia do Medão, Praias de Sesimbra, Sesimbra, São João do Estoril, São Pedro do Estoril

Pagico_063 Estudiosos da música indígena brasileira.

Scholars of indigenous Brazilian music.

Antonio Ruiz de Montoya, Heitor Villa-Lobos, Jean de Léry, Mário de Andrade

Pagico_063 Estudiosos da música indígena brasileira.

Scholars of indigenous Brazilian music.

Antonio Ruiz de Montoya, Heitor Villa-Lobos, Jean de Léry, Mário de Andrade

Pagico_063 Estudiosos da música indígena brasileira.

Scholars of indigenous Brazilian music.

Antonio Ruiz de Montoya, Heitor Villa-Lobos, Jean de Léry, Mário de Andrade

Pagico_085 Destinos turísticos do Brasil cuja temperatura no Inverno pode ser negativa.

Brazilian tourist destinations where the winter temperature can be negative.

Bento Gonçalves Rio Grande do Sul, Blumenau, Cambará do Sul, Campos do Jordão, Canela Rio Grande do Sul, Caxias do Sul, Curitiba, Florianópolis, Foz do Iguaçu, Gramado, Joinville, Londrina, Morro da Igreja, Parque Nacional da Serra Geral, Pico do Jabre, Planalto Serrano, Serra Catarinense, Serra Gaúcha, São José dos Ausentes, Terras Altas da Mantiqueira, Turismo em Santa Catarina

Pagico_086 Compositoras brasileiras de samba.

Brazilian samba songwriters.

Adriana Calcanhotto, Adryana Ribeiro, Beth Carvalho, Dolores Duran, Dona Ivone Lara, Elvira Pagã, Elza Soares, Leci Brandão, Marisa Monte, Marília Batista, Zélia Duncan

Pagico_088 Cidades portuguesas que têm festivais medievais.

Portuguese cities that have medieval festivals.

Aljubarrota, Chaves Portugal, Elvas, Fronteira Portugal, Mões, Santa Maria da Feira

Pagico_091 Estados fronteiriços de Moçambique.

Mozambican border-states.

Malawi, Suazilândia, Tanzânia, Zimbabwe, Zâmbia, África do Sul

Pagico_092 Cidades que fizeram parte do domínio português na Índia.

Cities that were part of the Portuguese Empire in India.

Baçaim, Bombaim, Calecute, Cananor, Cochim, Colombo, Coulão, Cranganor, Dadrá e Nagar-Aveli, Damão, Diu, Goa, Goa Velha, Malé, Mangalore, Masulipatão, Pangim, Ribandar, Surate, São Tomé de Meliapor, Vasco da Gama Goa

Pagico_094 Parques nacionais de Moçambique.

Mozambican national parks.

Parque Nacional da Gorongosa, Parque Nacional das Quirimbas, Parque Nacional do Bazaruto, Parque Nacional do Limpopo

Pagico_097 Escritores cabo-verdianos com obra publicada em crioulo.

Cape Verdean writers with published work in creole.

A. QUESTION DATASET

Baltasar Lopes da Silva, Eugénio Tavares, Gabriel Mariano, Ivone Ramos, Luís Romano de Madeira Melo, Manuel de Novas, Ovídio Martins, Sérgio Frusoni

Pagico_100 Ilhas de Moçambique.

Mozambican islands.

Arquipélago das Primeiras e Segundas, Arquipélago de Bazaruto, Bazaruto, Ilha de Moçambique, Ilha de Santa Carolina, Ilha de São Jorge Moçambique, Ilha de Xefina, Ilha do Ibo, Ilha dos Portugueses, Inhaca, Matemo, Quirimbas

Pagico_104 Pesquisadores do folclore brasileiro.

Brazilian folklore researchers.

Alfredo de Carvalho, Amadeu Amaral, Arthur Ramos, Augusto Meyer, Basílio de Magalhães, Canuto da Costa Azevedo, Celso de Magalhães, Emilia Biancardi, Franklin Cascaes, Gilberto Felisberto Vasconcellos, Glauco Saraiva, Gonçalves Fernandes, Gustavo Barroso, Heitor Villa-Lobos, José Vieira Couto de Magalhães, Lindolfo Gomes, Luciano Gallet, Luís da Câmara Cascudo, Marco Haurélio, Mário Pinto de Andrade, Mário de Andrade, Nereu do Vale Pereira, Oneida Alvarenga, Paixão Côrtes, Raul Lody, Saul Alves Martins, Sílvio Romero, Vicente Chermont de Miranda, Vicente Salles, Waldeloir Rego, Waldemar Henrique da Costa Pereira, Ático Vilas-Boas da Mota

Pagico_106 Vice-reis da Índia Portuguesa.

Viceroy of Portuguese India.

Afonso de Albuquerque, Afonso de Bragança, Duque do Porto, Afonso de Noronha, Aires de Saldanha, Antão de Noronha, António Luís Coutinho da Câmara, António de Melo e Castro, Bernardo José Maria Lorena e Silveira, Caetano de Melo e Castro, Constantino de Bragança, Diogo de Sousa, conde de Rio Pardo, Duarte de Meneses, Filipe de Mascarenhas, Francisco Coutinho, Francisco José de Sampaio e Castro, Francisco Teixeira da Silva, Francisco da Gama, Francisco de Almeida vice-rei da Índia, Francisco de Assis de Távora, Francisco de Mascarenhas, Francisco de Távora, Garcia de Noronha, Jerónimo de Azevedo, João Coutinho, João Nunes da Cunha, João da Silva Telo e Meneses, João de Castro, João de Saldanha da Gama, Luís Carlos Inácio Xavier de Meneses, Luís Mascarenhas, Luís de Ataíde, Luís de Mendonça Furtado e Albuquerque, Manuel Francisco Zacarias de Portugal e Castro, Manuel de Saldanha e

Albuquerque, Martim Afonso de Castro, Matias de Albuquerque, Miguel de Noronha, Pedro António de Meneses Noronha de Albuquerque, Pedro Mascarenhas (1470), Pedro Mascarenhas (1670), Pedro Miguel de Almeida Portugal e Vasconcelos, Pedro de Almeida, Rodrigo da Costa, Rui Lourenço de Távora, neto, Rui Lourenço de Távora, Vasco Fernandes César de Meneses, Vasco da Gama

Pagico_108 Jogadores de futebol nascidos em Cabo Verde que representaram a seleção portuguesa.

Football players born in Cape Verde who have represented the Portuguese national team.

Nani, Oceano da Cruz, Rolando Jorge Pires da Fonseca

Pagico_109 Candidatos a alguma das eleições presidenciais na Guiné-Bissau.

Candidates for any presidential elections in Guinea-Bissau.

Faustino Fudut Imbali, João Bernardo Vieira, Kumba Yalá, Malam Bacai Sanhá

Pagico_111 Padres católicos que estão ou estiveram ativos em Timor.

Catholic priests who are or were active in Timor.

Alberto Ricardo da Silva, Basílio do Nascimento, Carlos Filipe Ximenes Belo, Jaime Garcia Goulart, Miguel da Cruz Rangel

Pagico_112 Capitais das províncias de Angola.

The capitals of Angolan provinces.

Benguela, Cabinda cidade, Caxito, Huambo, Kuito, Lubango, Lucapa, Luenha Angola, M Banza Kongo, Malanje, Menongue, Namibe, Ondjiva, Saurimo, Serpa Pinto Angola, Sumbe, Sá da Bandeira Angola, Uíge

Pagico_116 Escritores lusófonos que passaram temporadas na prisão.

Lusophone writers who spent time in prison.

Agostinho Neto, Alves Redol, António José da Silva, Aquilino Ribeiro, Armindo José Rodrigues, Astrojildo Pereira, Camilo Castelo Branco, Carlos Coutinho, Chico Any-sio, Francisco Antunes Ferreira da Luz, Gerardo Melo Mourão, Graciliano Ramos, Henrique Abranches, Jaime Montestrela, José Luandino Vieira, José Manuel Tengarrinha, Luís Pereira Brandão, Luís de Camões, Manuel Alegre, Maria da Conceição

A. QUESTION DATASET

Vassalo e Silva da Cunha Lamas, Maurício Paiva de Lacerda, Ovídio Martins, Políbio Braga, Álvaro Cunhal

Pagico_118 Escritores moçambicanos que receberam o Prémio Camões.

Mozambican writers who have received The Camões Prize.

José Craveirinha

Pagico_124 Cabo-verdianos que participaram na guerra colonial na Guiné.

Cape Verdeans who participated in the colonial war in Guinea.

Amílcar Cabral, Aristides Maria Pereira, Pedro Pires

Pagico_128 Escritores portugueses que tenham vivido em Macau.

Portuguese writers who have lived in Macau.

Camilo Pessanha, Deolinda do Carmo Salvado da Conceição, José Rodrigues dos Santos, José Silveira Machado, José da Costa Nunes, Luís de Camões, Manuel Teixeira, Maria Ondina Braga, Venceslau de Moraes

Pagico_132 Deputados da FRELIMO.

FRELIMO's deputies.

Malangatana

Pagico_133 Futebolistas do Petro de Luanda.

Petro de Luanda players.

Antônio Lebo Lebo, Fabrice Alcebiades Maieco, Felix Katongo, José da Silva Santana Carlos, João Ricardo Pereira Batalha Santos Ferreira, Luís Delgado, Luís Mamona João Lamá, Paulo Batista Nsimba, Yamba Asha

Pagico_140 Cidades lusófonas conhecidas pelo seu Carnaval.

Lusophone cities known for their carnival celebrations.

Bissau, Caicó, Capim Branco, Carnaval do Rio de Janeiro, Elvas, Estarreja, Fortaleza, Funchal, Guapé, Lapão Bahia, Loulé, Loures, Luanda, Manaus, Mindelo Cabo Verde, Nova Ponte, Olinda, Ovar, Porto Alegre, Recife, Rio de Janeiro cidade, Salvador Bahia, Sesimbra, Sines, São Paulo cidade, Torres Vedras, Uruguaiana, Vitória Espírito Santo

Pagico_149 Arquitetos de países lusófonos com obras em países estrangeiros na América do Norte e na Europa.

Architects from lusophone countries with works in foreign countries in North America and Europe.

Gonçalo Byrne, Lúcio Costa, Oscar Niemeyer, Álvaro Siza Vieira

Pagico_153 Toureiros a cavalo de países lusófonos com carreira internacional.

Internationally-known bullfighters on horseback from lusophone countries.

António Ribeiro Telles, José Mestre Baptista, João Branco Nuncio

A.2 QALD Dataset

Below, you may find the subset of questions from QALD Competition and their correct answers that were used in used in our experiments.

QALD_010 *In which country does the Nile start?*

Em qual país começa o Rio Nilo?

Rwanda, Ethiopia

QALD_028 *Give me all communist countries.*

Liste todos paises comunistas.

Republic of China, Republic of Cuba, Lao Peoples Democratic Republic, Socialist Republic of Vietnam

QALD_032 *Which countries adopted the Euro?*

Quais são os paises que adotaram o euro?

Austria, Belgium, Cyprus, Estonia, Finland, France, Germany, Greece, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Portugal, Slovakia, Slovenia, Spain

QALD_036 *Through which countries does the Yenisei river flow?*

Por quais paises o rio Yenisei corre?

Mongolia, Russia

A. QUESTION DATASET

QALD_062 *Who created Wikipedia?*

Quais são os criadores da Wikipedia?

Jimmy Wales, Larry Sanger

QALD_074 *Which capitals in Europe were host cities of the summer olympic games?*

Quais as capitais na Europa que hospedaram o Jogos Olímpicos de Verão?

Amsterdam, Athens, Berlin, Helsinki, London, Moscow, Paris, Rome, Stockholm

QALD_114 *Give me all members of Prodigy.*

Cite todos membros do Prodigy.

Liam Howlett, Keith Flint, Maxim Reality

QALD_141 *Who founded Intel?*

Quais são os fundadores da Intel?

Robert Noyce, Gordon Moore

QALD_155 *Which Greek goddesses dwelt on Mount Olympus?*

Quais Deusas gregas moravam no Monte Olimpo?

Aphrodite, Athena, Hera, Eileithyia, Hygieia, Hebe (mythology), Nike (mythology)

QALD_176 *List the children of Margaret Thatcher.*

Liste os filhos de Margaret Thatcher.

Carol Thatcher, Mark Thatcher

A.3 Temporal Dataset

Below, you may find the subset of questions their correct answers that were used in used in our experiments about Temporal List Questions.

Pagico_008 Telenovelas brasileiras passadas no tempo da escravatura no Brasil.

Brazilian soap operas set in the time of slavery in Brazil.

A Escrava Isaura (2004), Banzo, Dona Beija, Escrava Isaura (1976), Força de um Desejo, Helena (1952), Helena (1975), Helena (1987), Pacto de Sangue, Paixões Proibidas, Sangue do Meu Sangue, Sangue do Meu Sangue (1969), Sangue do Meu Sangue (1995), Sinhazinha Flô, Sinhá Moça (1986), Sinhá Moça (2006), Xica da Silva telenovela

Pagico_034 Viajantes ou exploradores que escreveram sobre o Brasil do século XVI.

Travellers or explorers who wrote about Brazil in the 16th century.

André Thévet, Binot Paulmier de Gonneville, Gabriel Soares de Sousa, Gaspar de Carvajal, Hans Staden, James Lancaster, Jean de Léry, Pero Vaz de Caminha

Pagico_050 Jornais que circularam no Rio de Janeiro entre 1910 e 1960.

Newspapers in circulation in Rio de Janeiro between 1910 and 1960.

Correio da Manhã Brasil, Diário Carioca, Diário da Noite Rio de Janeiro, Jornal das Moças, Jornal do Brasil, Jornal do Commercio, Monitor Campista, Mundo Sportivo, O Fluminense, O Globo, O Paiz, Tribuna da Imprensa, Tribuna de Petrópolis, Última Hora

Pagico_068 Bandas brasileiras de punk formadas até 1980 em São Paulo.

Brazilian punk bands formed before 1980 in São Paulo.

AI-5 banda, Condutores de Cadáver, Cólera banda, DZK, Lixomania, Olho Seco, Ratos de Porão, Restos de Nada, Ulster banda

Pagico_078 Escritoras de língua portuguesa que tenham publicado livros para crianças entre 1850 e 1940.

Female portuguese authors who published childrens' books between 1850 and 1940.

Ana de Castro Osório, Maria Amália Vaz de Carvalho, Maria da Conceição Vassalo e Silva da Cunha Lamas

TP_001 Quais eram os partidos politicos existentes antes de 1964?

Which were the political parties that existed before 1964?

Partido Comunista Brasileiro, Partido Trabalhista Nacional, Partido Trabalhista Brasileiro, Partido Socialista Brasileiro, Partido Comunista do Brasil, Partido do Movimento Democrático Brasileiro, Alianca Renovadora Nacional

TP_002 Quem ganhou o premio Nobel entre 1900 a 1920?

Who won the Nobel prize between 1900 and 1920?

Adolf von Baeyer, Albert Abraham Michelson, Albrecht Kossel, Alexis Carrel, Alfred Hermann Fried, Alfred Werner, Allvar Gullstrand, August Krogh, Auguste Marie , Bertha von Suttner, Bjørnstjerne Bjørnson, Camillo Golgi, Carl Spitteler, Charles Albert Gobat, Charles Édouard Guillaume, Charles Glover Barkla, Charles Louis

A. QUESTION DATASET

Alphonse Laveran, Charles Richet, Eduard Buchner, Élie Ducommun, Elihu Root, Emil Adolf von Behring, Emil Theodor Kocher, Ernest Rutherford, Ernesto Teodoro Moneta, François Beernaert, Frédéric Mistral, Frédéric Passy, Fredrik Bajer, Fritz Haber, Gabriel Lippmann, Gerhart Hauptmann, Giosuè Carducci, Guglielmo Marconi, Gustaf Dalén, Heike Kamerlingh Onnes, Hendrik Lorentz, Henri Becquerel, Henri La Fontaine, Henri Moissan, Henrik Pontoppidan, Henryk Sienkiewicz, Hermann Emil Fischer, Ilya Ilyich Mechnikov, Ivan Pavlov, J. J. Thomson, Jacobus Henricus van 't Hoff, Johannes Diderik van der Waals, Johannes Stark, John William Strutt, José Echegaray, Jules Bordet, Karl Adolph Gjellerup, Karl Ferdinand Braun, Klas Pontus Arnoldson, Knut Hamsun, Louis Renault, Marie Curie, Maurice Maeterlinck, Max Planck, Max von Laue, Niels Ryberg Finsen, Otto Wallach, Paul Ehrlich, Paul Heyse, Paul Sabatier, Paul-Henri-Benjamin d'Estournelles de Constant, Philipp Lenard, Pierre Curie, Pieter Zeeman, Rabindranath Tagore, Richard Willstätter, Robert Bárány, Robert Koch, Romain Rolland, Ronald Ross, Rudolf Christoph Eucken, Rudyard Kipling, Santiago Ramón y Cajal, Selma Lagerlöf, Sully Prudhomme, Svante Arrhenius, Theodor Mommsen, Theodore Roosevelt, Theodore William Richards, Tobias Asser, Verner von Heidenstam, Victor Grignard, Walther Nernst, Wilhelm Ostwald, Wilhelm Röntgen, Wilhelm Wien, William Henry Bragg, William Lawrence Bragg, William Ramsay, William Randal Cremer, Woodrow Wilson

TP_003 Quais foram as novelas brasileiras dos últimos 5 anos?

Which were the Brazilian novels from the last 5 years?

Em família, Amor à vida, Salve Jorge, Avenida Brasil, Fina Estampa, Insensato Coração, Passione, Viver a vida, Caminho das Índias, A Favorita

TP_004 Que países boicotaram os Jogos Olímpicos de 1980?

Which countries boycotted the 1980 Olympic Games?

Albânia, Antilhas Holandesas, Argentina, Bahamas, Brunei, Bangladesh, Barbados, Belize, Bermudas, Bolívia, Canadá, Ilhas Cayman, República Centro-Africana, Chade, Chile, China, Taipé Chinesa, Costa do Marfim, Egito, El Salvador, Fiji, Gabão, Gâmbia, Gana, Haiti, Honduras, Hong Kong, Indonésia, Irã, Israel, Japão, Quênia, Coreia do Sul, Libéria, Liechtenstein, Malawi, Malásia, Maurícia, Mónaco, Marrocos, Antilhas Holandesas, Níger, Noruega, Paquistão, Panamá, Papua-Nova Guiné, Paraguai, Filipinas, Catar, Arábia Saudita, Singapura, Somália, Sudão, Suriname, Suazilândia,

Tailândia, Togo, Tunísia, Turquia, Emirados Árabes Unidos, Estados Unidos, Uruguai, Ilhas Virgens Americanas, Alemanha Ocidental, Zaire

TP_005 Quais são as bandas brasileiras de rock dos anos 80?

Which were the Brazilian rock bands from the 1980s?

Aborto Elétrico, Barão Vermelho, Biquini Cavadão, Blitz, Brylho, Camisa de Vênus, Capital Inicial, Cascavelettes, Defalla, Engenheiros do Hawaii, Fausto Fawcett e os robôs efêmeros, Garotos da rua, Garotos Podres, Hanoi Hanoi, Heróis da Resistência, Herva Doce, Inimigos do Rei, Ira, João Penca e os seus miquinhos amestrados, Kid Abelha, Legião Urbana, Lobão e os Ronaldos, Nenhum de nós, Paralamas do Sucesso, Plebe Rude, Rádio Taxi, Replicantes, Roupas novas, Rpm, Titãs, Tóquio, Ultraje a rigor, Cazuza, Eduardo Dusek, Ed Motta, Guilherme Arantes, Kiko Zambianchi, Léo Jaime, Lobão, Lulu Santos, Marina, Ritchie, Angra

B

LX-ListQuestion Answers

Below we show the output of the LX-ListQuestion QA system to the question dataset used in the experiments described in Chapter 5 - Evaluation.



Figure B.1: LX-ListQuestion QA system answering the question Pagico_004

B. LX-LISTQUESTION ANSWERS

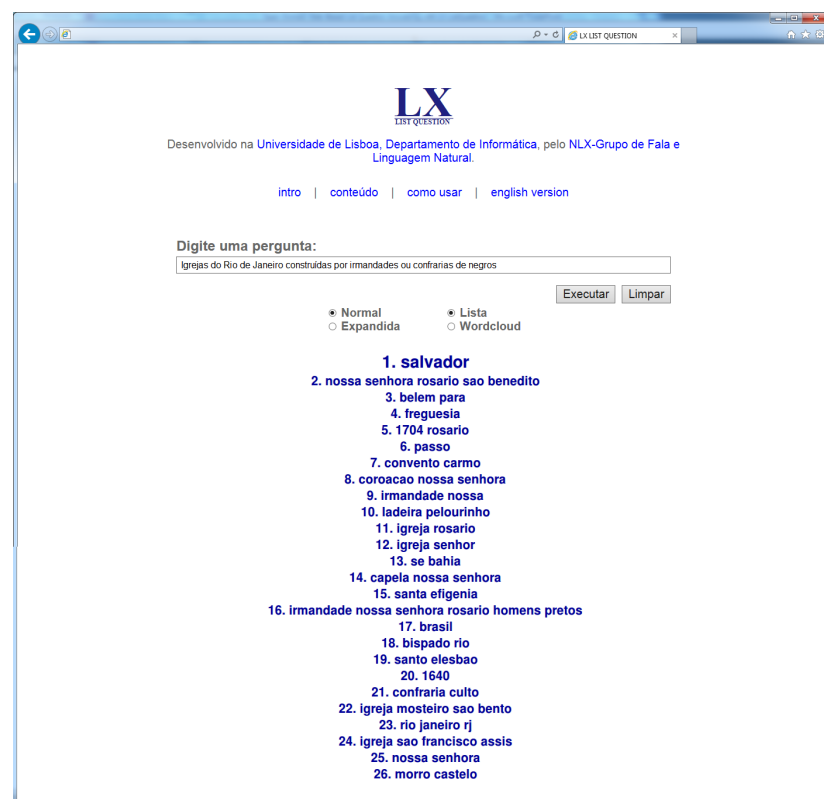


Figure B.2: LX-ListQuestion QA system answering the question Pagico_054



Figure B.3: LX-ListQuestion QA system answering the question Pagico_062

B. LX-LISTQUESTION ANSWERS



Figure B.4: LX-ListQuestion QA system answering the question Pagico_086



Figure B.5: LX-ListQuestion QA system answering the question Pagico_088

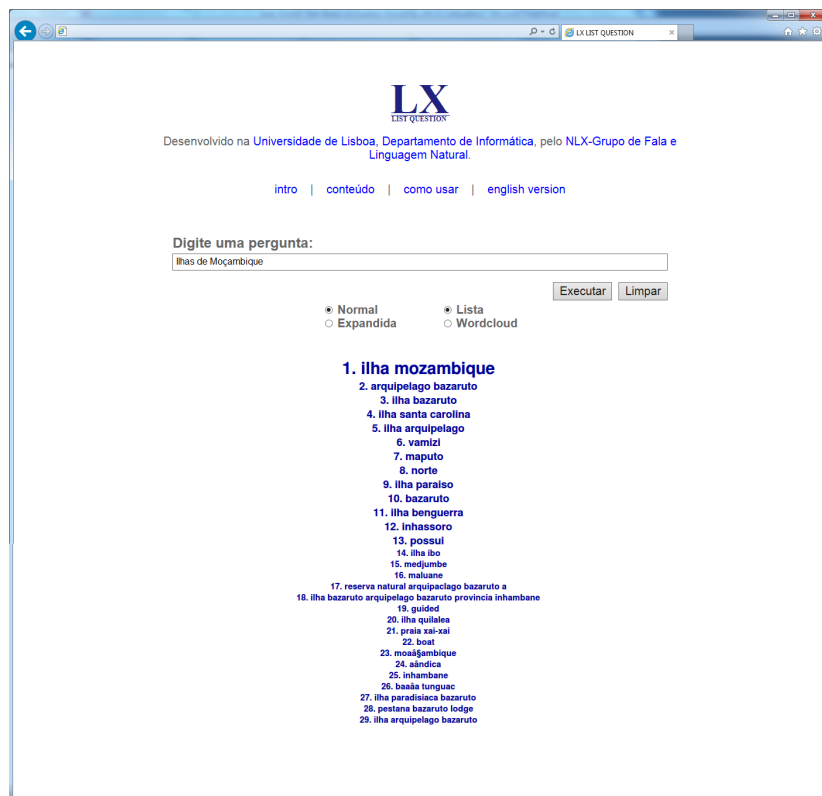


Figure B.6: LX-ListQuestion QA system answering the question Pagico_100



Figure B.7: LX-ListQuestion QA system answering the question Pagico_109

B. LX-LISTQUESTION ANSWERS



Figure B.8: LX-ListQuestion QA system answering the question Pagico_112



Figure B.9: LX-ListQuestion QA system answering the question Pagico_133



Figure B.10: LX-ListQuestion QA system answering the question Pagico_140

B. LX-LISTQUESTION ANSWERS



Figure B.11: LX-ListQuestion QA system answering the question QALD_010



Figure B.12: LX-ListQuestion QA system answering the question QALD_028

B. LX-LISTQUESTION ANSWERS

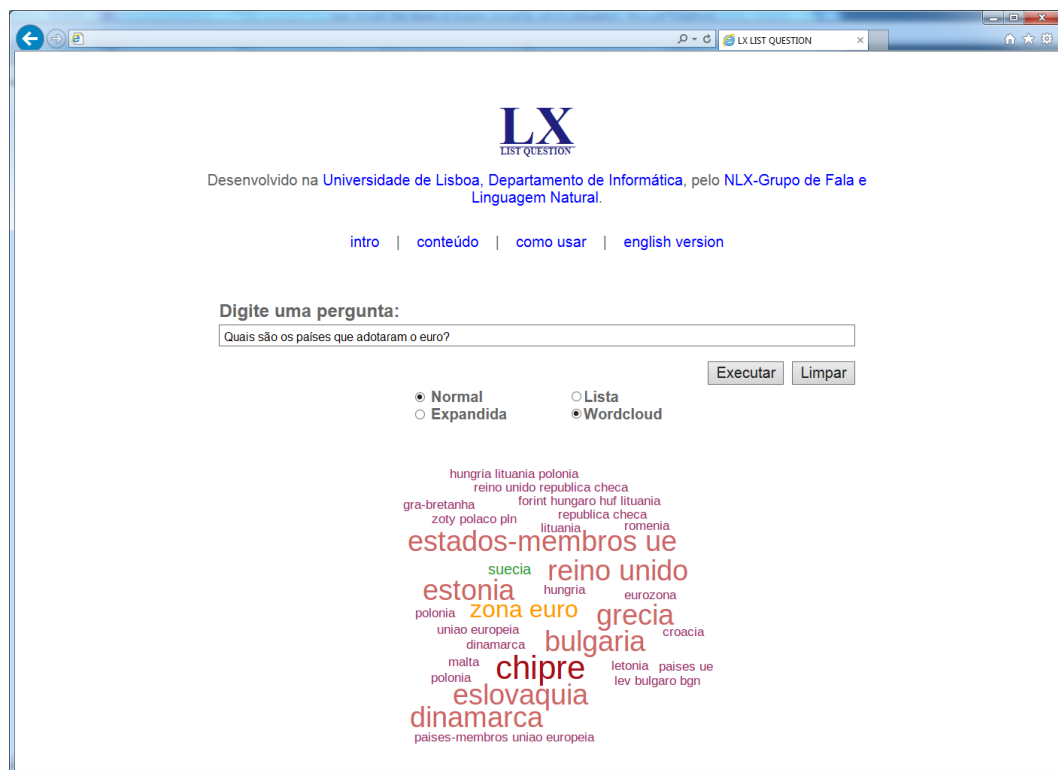


Figure B.13: LX-ListQuestion QA system answering the question QALD_032



Figure B.14: LX-ListQuestion QA system answering the question QALD_036

B. LX-LISTQUESTION ANSWERS

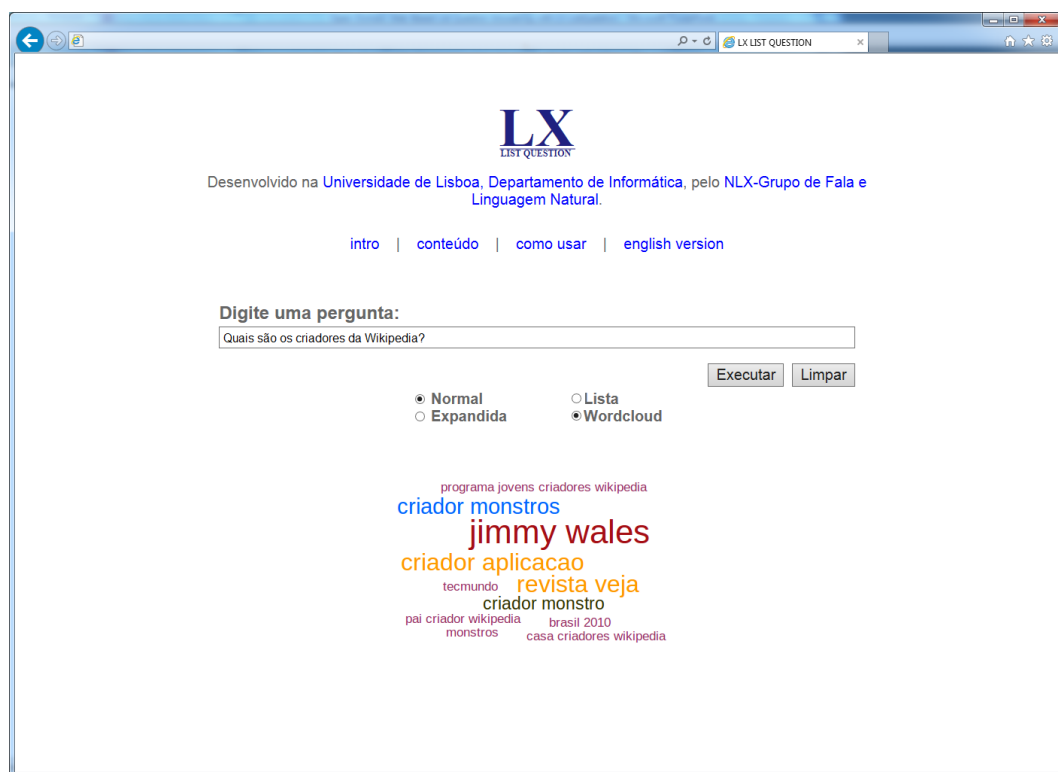


Figure B.15: LX-ListQuestion QA system answering the question QALD_062



Figure B.16: LX-ListQuestion QA system answering the question QALD_074

B. LX-LISTQUESTION ANSWERS

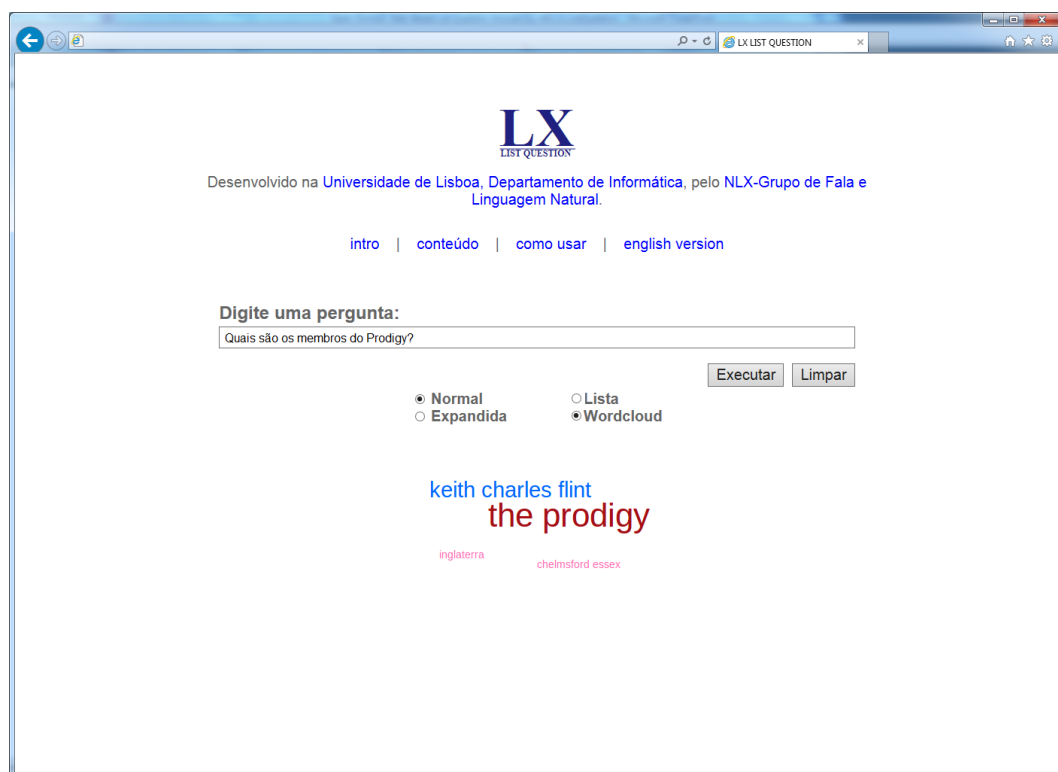


Figure B.17: LX-ListQuestion QA system answering the question QALD_114

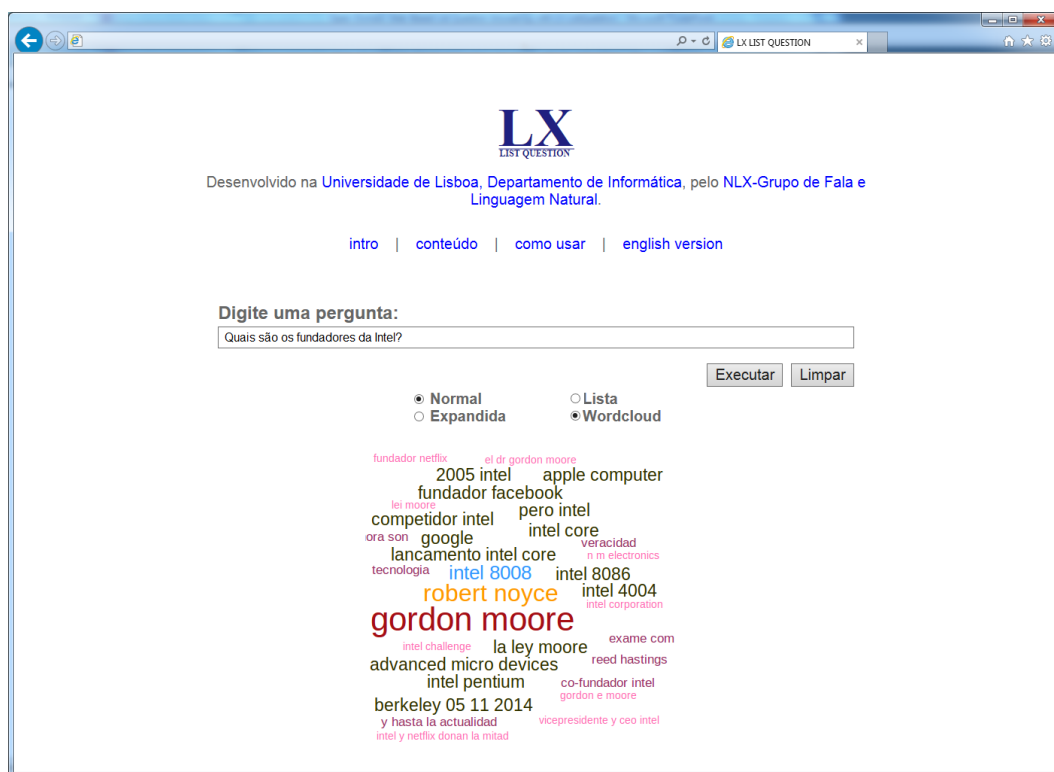


Figure B.18: LX-ListQuestion QA system answering the question QALD_141

B. LX-LISTQUESTION ANSWERS



Figure B.19: LX-ListQuestion QA system answering the question QALD_155

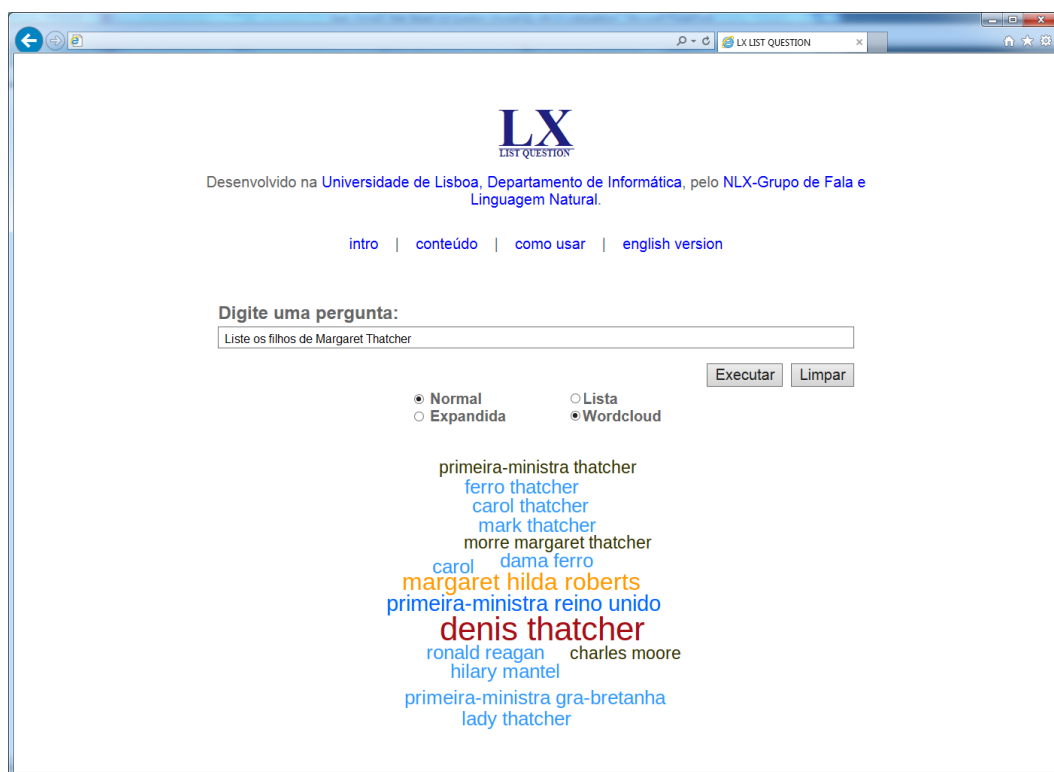


Figure B.20: LX-ListQuestion QA system answering the question QALD_176

B. LX-LISTQUESTION ANSWERS

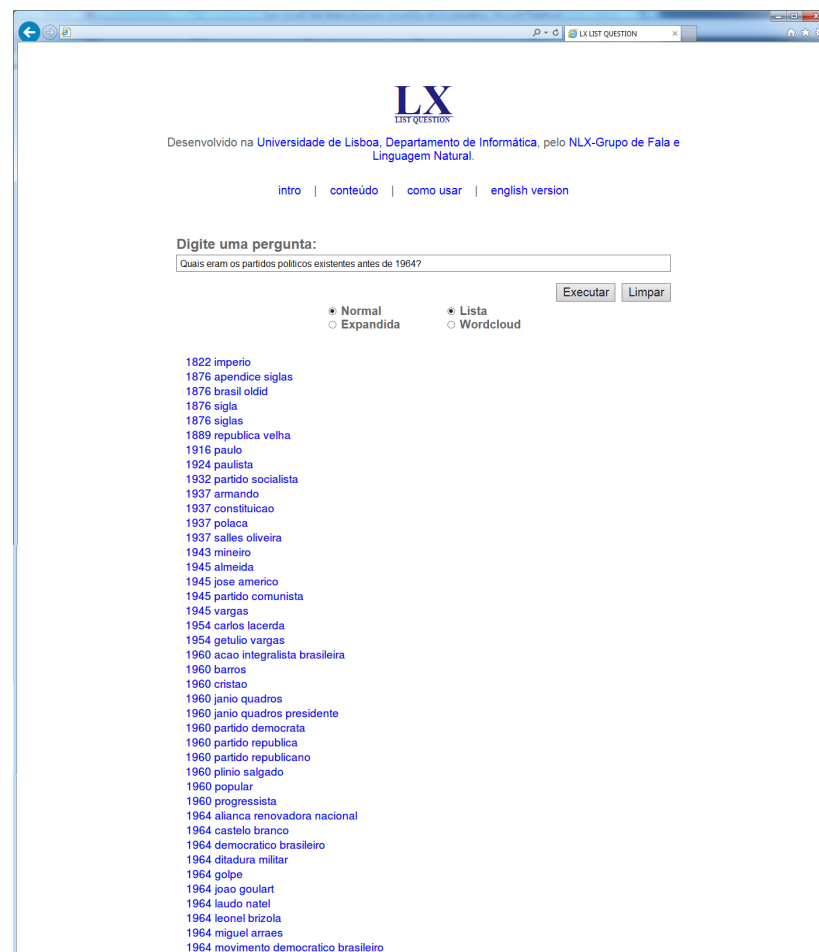


Figure B.21: LX-ListQuestion QA system answering the question TP_001

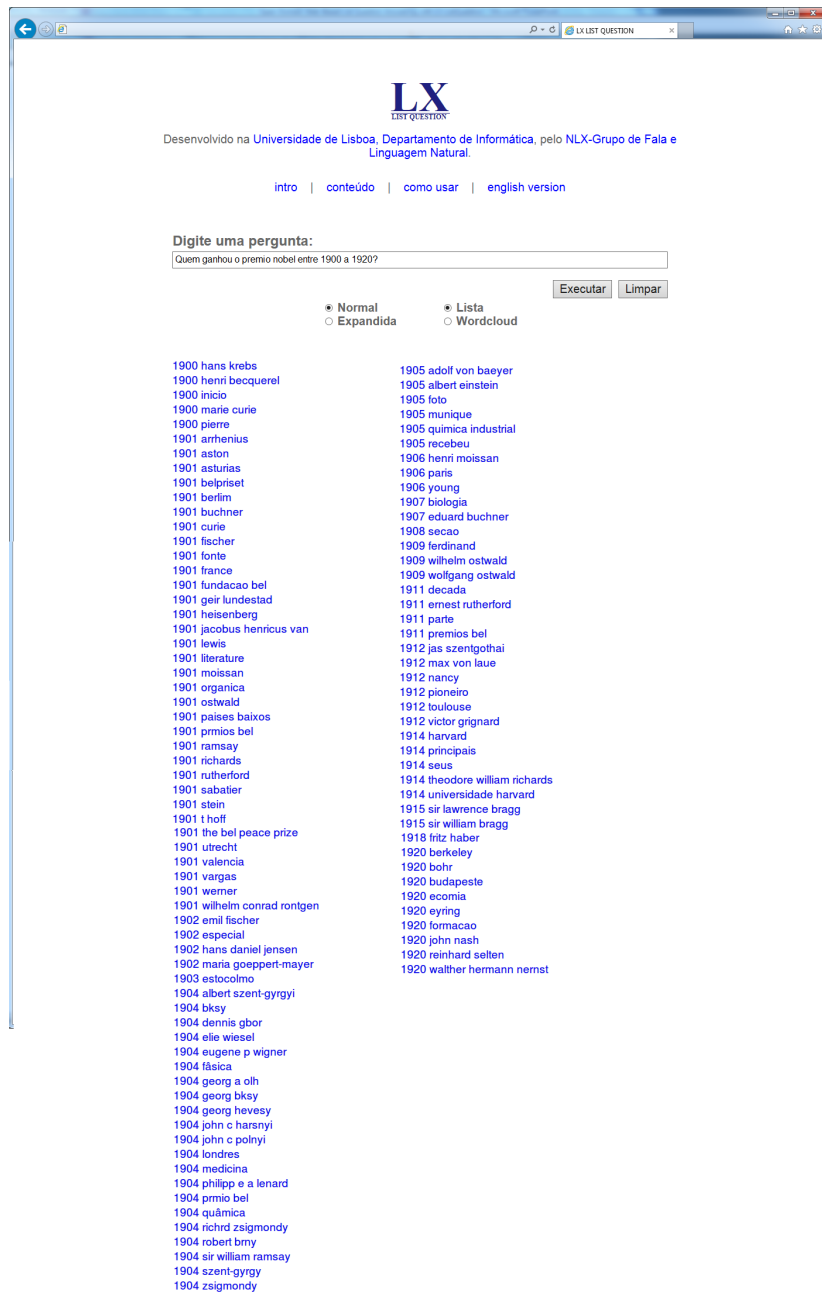


Figure B.22: LX-ListQuestion QA system answering the question TP_002

B. LX-LISTQUESTION ANSWERS



Figure B.23: LX-ListQuestion QA system answering the question TP_003

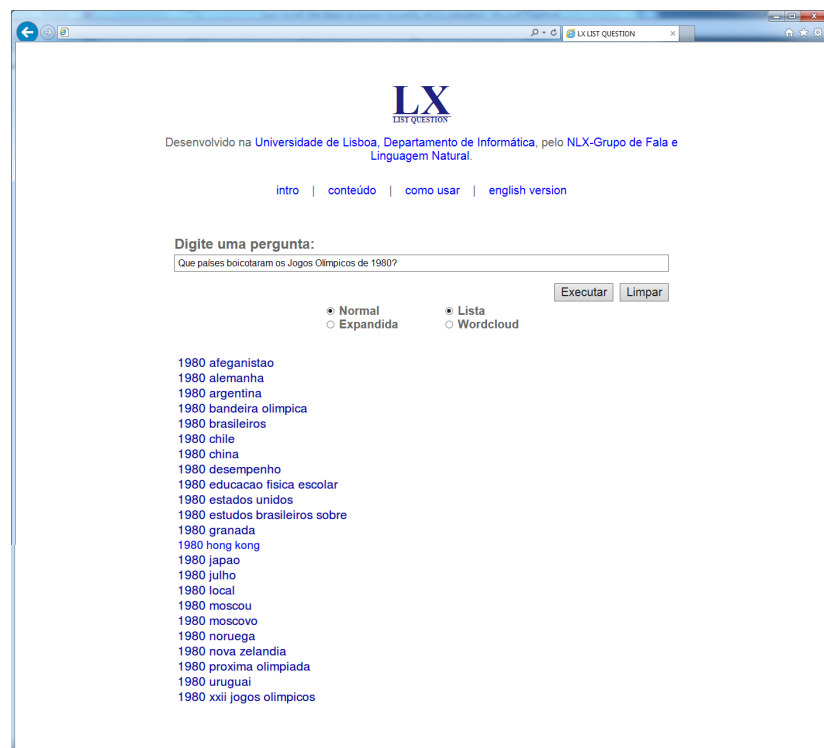


Figure B.24: LX-ListQuestion QA system answering the question TP_004



Figure B.25: LX-ListQuestion QA system answering the question TP_005



XisQuê Answers

Below we show the output of the XisQuê QA system to the question dataset used in the experiments described in Chapter 5 - Evaluation.

XisQuê
Question answering
for the Portuguese Web

Enter a question:

Ask Clear

Question: Mulheres violoncelistas de língua portuguesa

Answer #1:
língua portuguesa
[source document](#)

Answer #2:
Língua Portuguesa: Perguntas e Respostas!
[source document](#)

Answer #3:
Ciberdúvidas da Língua Portuguesa
[source document](#)

Answer #4:
«Começou a ter formação musical aos seis anos – tanto ela como os oito irmãos. Na casa da família, essa aprendizagem era tão importante como a matemática, a língua portuguesa ou qualquer outra disciplina do currículo pedagógico.
[source document](#)

Answer #5:
Estudava e praticava muito tendo sido uma das primeiras mulheres violoncelistas profissionais a tocar a solo.
[source document](#)

Figure C.1: XisQuê QA system answering the question Pagico_004

C. XISQUÊ ANSWERS

XisQuê Enter a question:
Question answering for the Portuguese Web

Question: Igrejas do Rio de Janeiro construídas por irmandades ou confrarias de negros.

Answer #1:
FRIDMAN, F.; MACEDO, V. L. Irmandades de negros: desterro no Rio de Janeiro colonial.
[source document](#)

Answer #2:
Guia das igrejas históricas da cidade do Rio de Janeiro.
[source document](#)

Answer #3:
Barroco e rococó nas igrejas do Rio de Janeiro.
[source document](#)

Answer #4:
O Rio de Janeiro e suas igrejas.
[source document](#)

Figure C.2: XisQuê QA system answering the question Pagico_054

XisQuê Enter a question:
Question answering for the Portuguese Web

Question: Praias de Portugal boas para a prática de surf

Answer #1:
Onda para surfistas de todos os níveis. Quando o vento está de noroeste, a praia do Guincho torna-se um dos locais mais ventosos de Portugal, com boas condições para a prática de windsurf e kitesurf.
[source document](#)

Answer #2:
Historial O Guincho é uma das praias onde se pratica surf em portugal desde os anos 60.
[source document](#)

Answer #3:
Portugal está também representado na lista da Condé Nast pelas praias de Peniche, descritas como "um paraíso do surf com maiúsculas" para desfrutar do Verão. "Se é a primeira vez que sobe a uma prancha, as escolas Peniche Surf Camp ensinam-lhe o básico e numa semana estará a 'surfar' com toda a naturalidade", frisa a revista.
[source document](#)

Answer #4:
Clique AQUI para aceder à compilação das melhores praias para a prática de surf da Condé Nast (em espanhol).
[source document](#)

Answer #5:
Boas Notícias - Praias portuguesas entre "paraísos" europeus do surf
[source document](#)

Figure C.3: XisQuê QA system answering the question Pagico_062

XisQuê
Question answering
for the Portuguese Web

Enter a question:

Ask Clear

Question: Compositoras brasileiras de samba

Answer #1:

Exatamente nesse ano de 1959 em que a bossa nova se consolidava definitivamente como a mais dramática renovação da música popular brasileira de todos os tempos, morria a cantora e compositora Dolores Duran, uma de suas precursoras, também considerada das melhores compositoras brasileiras de todos os tempos, e a cantora de múltiplas facetas, que cantava todos os ritmos possíveis e em diversos idiomas.

[source document](#)

Answer #2:

Logo em seguida, ainda pelo mesmo selo, voltaria a gravar outro 78 rpm com músicas brasileiras, interpretando Outono, um samba de um futuro parceiro, Billy Blanco, e Um Amor Assim, samba-canção da cantora e compositora Dora Lopes.

[source document](#)

Answer #3:

TAGS: mulheres artistas compositoras compositoras de samba mulheres interpretes mulheres compositoras A primeira compositora de escola de samb mulheres no samba negras compositoras negras mulheres negras site Raíza Brasil revista Raíza Brasil

[source document](#)

Answer #4:

Neste contexto, a indústria fonográfica brasileira estimulou uma nova roupagem comercial do samba-canção, adaptado às influências musicais vindas de fora e bastante tocadas nas rádios brasileiras no período.

[source document](#)

Figure C.4: XisQuê QA system answering the question Pagico_086

XisQuê
Question answering
for the Portuguese Web

Enter a question:

Ask Clear

Question: Cidades portuguesas que têm festivais medievais

Answer #1:

- "Torres Vedras", in Atlas de cidades medievais portuguesas, coord.

[source document](#)

Answer #2:

Cidades portuguesas t?m festivais medievais - Ask.com YouTube Search

[source document](#)

Figure C.5: XisQuê QA system answering the question Pagico_088

C. XISQUÊ ANSWERS

XisQuê Enter a question:

Question answering
for the Portuguese Web

Ask Clear

Question: Ilhas de Moçambique

Answer #1:
Moçambique - Ilhas...
[source document](#)

Answer #2:
O Commons possui uma categoria contendo imagens e outros ficheiros sobre Ilhas de Moçambique
[source document](#)

Answer #3:
Obtida de "http://pt.wikipedia.org/wiki?title=Categoria:Ilhas_de_Moçambique&oldid=552398"
[source document](#)

Answer #4:
Medjumbe Moçambique - Episódios - Ilhas Paradisiacas - Canal Off
[source document](#)

Answer #5:
Categoria:Ilhas de Moçambique - Ask.com Encyclopedia
[source document](#)

Figure C.6: XisQuê QA system answering the question Pagico_100

XisQuê Enter a question:

Question answering
for the Portuguese Web

Ask Clear

Question: Candidatos a alguma das eleições presidenciais na Guiné-Bissau

Answer #1:
candidatos, Eleições Presidenciais, Guiné-Bissau
[source document](#)

Answer #2:
Nove candidatos já entregaram no Supremo Tribunal de Justiça (STJ) da Guiné-Bissau a documentação exigida por lei para concorrer às eleições presidenciais de 13 de Abril, disse hoje à Lusa fonte do órgão.
[source document](#)

Answer #3:
- 28 de março: A Comissão Nacional de Eleições (CNE) da Guiné-Bissau considera improcedentes as reclamações de fraude apresentadas por cinco candidatos nas eleições presidenciais.
[source document](#)

Answer #4:
Na altura Carlos Gomes Júnior, primeiro-ministro, tinha sido o candidato mais votado na primeira volta e era suposto vir a medir forças com Kumba Yalá, ex chefe de Estado (este faleceu entretanto no mês transacto, poucos dias antes da primeira volta das eleições presidenciais em curso).
[source document](#)

Answer #5:
Estas eleições inscrevem-se no regresso à ordem constitucional, exigido pela comunidade internacional, após o golpe de Estado de 12 de Abril de 2012 que impediu a segunda volta das eleições presidenciais.
[source document](#)

Figure C.7: XisQuê QA system answering the question Pagico_109

XisQuê

Resposta a perguntas
na Web portuguesa

Introduza uma pergunta:

Perguntar

Limpar

Pergunta: Capitais das províncias da Angola.

Resposta #1:

O primeiro mostra a divisão política do país com suas 18 províncias, capitais e cidades.

[documento fonte](#)

Resposta #2:

Foi um "parto" longo e difícil para a existência de facto da Bolsa, porquanto os trabalhos que resultaram na institucionalização da Comissão de Mercado de Capitais (CMC) tiveram início em 1998, com a produção dos primeiros estudos sobre a formalização de uma Bolsa de Valores de Angola e culminaram com a aprovação do Decreto 9/05, de 18 de Maio de 2005, e na aprovação da Lei de 12/05, de 23 de Setembro, e na Lei de 13/05, de 30 de Setembro.

[documento fonte](#)

Resposta #3:

Por outro lado, os bancos continuaram com a sua estratégia de expansão da actividade às zonas mais recônditas dos centros das cidades capitais das províncias, de modo a apoiar o processo de dinamização da economia destas localidades.

[documento fonte](#)

Resposta #4:

Província de Luanda

[documento fonte](#)

Figure C.8: XisQuê QA system answering the question Pagico_112

XisQuê

Question answering
for the Portuguese Web

Enter a question:

Ask

Clear

Question: Futebolistas do Petro de Luanda

Answer #1:

Futebolistas Petro Luanda - Ask.com YouTube Search

[source document](#)

Answer #2:

(2) O Atlético Luanda virou Atlético Petróleos de Luanda (Petro Atlético).

[source document](#)

Answer #3:

Atlético Petróleos de Luanda, conhecido também como Petro Atlético de Luanda, Petro Atlético, ou simplesmente Petro de Luanda, é um clube tradicional de futebol de Luanda, Angola, fundado em 1980. O clube ganhou o seu primeiro título, a Liga Angolana de 1982.

[source document](#)

Answer #4:

Home » FIFA » CAF » Angola » Petro Luanda

[source document](#)

Answer #5:

Petro Luanda

[source document](#)

Figure C.9: XisQuê QA system answering the question Pagico_133

C. XISQUÊ ANSWERS

XisQuê Enter a question:

Question answering
for the Portuguese Web

Question: Cidades lusófonas conhecidas pelo seu Carnaval

Answer #1:
O ritual de embarque de dom Carnaval, em escaler que saía de um dos cais do rio Capibaribe – paródia das entradas solenes nas cidades, dos reis e importantes autoridades das sociedades de Antigo Regime –, e que passou a ser realizado todos os anos pelos sócios do clube, introduzia os cortejos de rua nas festas de Aleluia.
[source document](#)

Answer #2:
As cidades apresentam grande disparidade, seja os recalcados de Jaboatão, a terceira cidade de Pernambuco, ou Olinda, a cidade dormitório, que só serve durante os dias de carnaval, depois volta a ser a bostinha de sempre.
[source document](#)

Figure C.10: XisQuê QA system answering the question Pagico_140

XisQuê Enter a question:

Question answering
for the Portuguese Web

Question: Quais eram os partidos políticos existentes antes de 1964

Answer #1:
O golpe militar de 1964 não acabou imediatamente com os partidos políticos existentes, muito embora o primeiro dos Atos Institucionais tenha sido acompanhado por uma lista de cassações que levou vários políticos ao exílio.
[source document](#)

Answer #2:
1965 Extinguem-se os partidos políticos existentes e institui-se o bipartidarismo, com a Aliança Renovadora Nacional (Arena), de apoio ao governo, e o MDB (Movimento Democrático Brasileiro), de oposição.
[source document](#)

Answer #3:
-Extinguem-se em 1965 os partidos políticos existentes e institui-se o bipartidarismo, com a Aliança Renovadora Nacional (Arena), de apoio ao governo, e o MDB (Movimento Democrático Brasileiro), de oposição.
[source document](#)

Answer #4:
Ato Institucional n° 2 (27/10/65) – dissolveu os partidos políticos existentes e criou o bipartidarismo.
[source document](#)

Answer #5:
Este ato proibiu os partidos políticos existentes (encerrando o sistema partidário inaugurado em 1945) e deixou espaço para a formação de apenas dois partidos – que acabaram sendo a Aliança Renovadora Nacional (ARENA, partido oficial) e o Movimento Democrático Brasileiro (MDB, oposição consentida).

Figure C.11: XisQuê QA system answering the question TP_001



Question: Quem ganhou o premio Nobel entre 1900 e 1920?

Figure C.12: XisQuê QA system answering the question TP_002



Question: Quais foram as novelas brasileiras dos últimos 5 anos?

Answer #1:
foram novelas brasileiras?ltimos 5 anos - Ask.com YouTube Search
[source document](#)

Answer #2:
Adital - TRÁFICO DE PESSOAS: nos últimos 15 anos, apenas cinco casos foram julgados no Ceará
[source document](#)

Answer #3:
Top Listas das Celebidades: Confira 10 novelas de maior sucesso nos últimos 10 anos
[source document](#)

Answer #4:
Confira 10 novelas de maior sucesso nos últimos 10 anos
[source document](#)

Answer #5:
"Foi uma novela marcante, um grande sucesso de audiência e repercussão. [Nos últimos dez anos.] houve uma mudança no cenário. Hoje em dia, a concorrência com outras mídias é muito maior. Vivemos um outro tempo, diferente de dez anos atrás", analisa Nilson Xavier, crítico de novelas e autor do Almanaque da Telenovela Brasileira.
[source document](#)

Figure C.13: XisQuê QA system answering the question TP_003

C. XISQUÊ ANSWERS

XisQuê Enter a question:

Question answering
for the Portuguese Web

Ask Clear

Question: Que países boicotaram os Jogos Olímpicos de 1980?

Answer #1:
Site consultado: < <http://www.educacaofisica.com.br/index.php/eventos/eventos-2012/olimpiadas-2012/22142-xxii-jogos-olimpicos-da-era-moderna-moscou-1980> > .
[source document](#)

Figure C.14: XisQuê QA system answering the question TP_004

XisQuê Enter a question:

Question answering
for the Portuguese Web

Ask Clear

Question: Quais são as bandas brasileiras de rock dos anos 80?

Answer #1:
s?o bandas brasileiras rock anos 80 - Ask.com YouTube Search
[source document](#)

Answer #2:
Resgate é uma banda de rock cristão brasileira, formada na cidade de São Paulo em 5 de Maio de 1989, estando há vinte e quatro anos em atividade com a mesma formação. Apesar de iniciar suas atividades ainda no final da década de 80, despontou no meio cristão na década seguinte, época em que as bandas Oficina G3, Fruto Sagrado, Katsbarnea e Catedral alcançavam grande notoriedade no meio gospel.
[source document](#)

Answer #3:
Nesta década, algumas bandas que fizeram fama nos anos 80 continuaram fazendo sucessos como Legião Urbana, Paralamas do Sucesso, Titãs e Kid Abelha e outras bandas surgiram adicionando mais diversidade ao rock como Raimundos, Charlie Brown Jr, Chico Science & Nação Zumbi, entre outros.
[source document](#)

Answer #4:
Evandro Mesquita, Lobão, Patrícia Travassos e Fernanda Abreu são os principais nomes dessa banda que marcou a história do rock dos anos 80, com irreverência e atitude.
[source document](#)

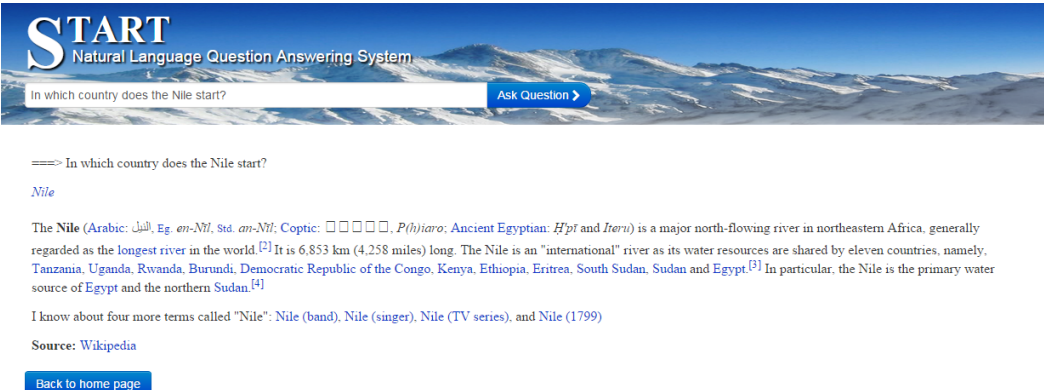
Answer #5:
Legião faz rock anos 80 que é uma das piores coisas que já existiu!
[source document](#)

Figure C.15: XisQuê QA system answering the question TP_005

D

START Answers

Below we show the output of the START QA system to the question dataset used in the experiments described in Chapter 5 - Evaluation.



START
Natural Language Question Answering System

In which country does the Nile start? [Ask Question](#)

==> In which country does the Nile start?

Nile

The **Nile** (Arabic: النيل, *Eg. en-Nīl, Std. an-Nīl; Coptic: ⲁⲓⲛⲓⲗ, P(h)jiaro; Ancient Egyptian: Ḥꜣꜣt and Iteru*) is a major north-flowing river in northeastern Africa, generally regarded as the **longest river** in the world.^[2] It is 6,853 km (4,258 miles) long. The Nile is an "international" river as its water resources are shared by eleven countries, namely, [Tanzania](#), [Uganda](#), [Rwanda](#), [Burundi](#), [Democratic Republic of the Congo](#), [Kenya](#), [Ethiopia](#), [Eritrea](#), [South Sudan](#), [Sudan](#) and [Egypt](#).^[3] In particular, the Nile is the primary water source of Egypt and the northern Sudan.^[4]

I know about four more terms called "Nile": Nile (band), Nile (singer), Nile (TV series), and Nile (1799)

Source: [Wikipedia](#)

[Back to home page](#)

Figure D.1: START QA system answering the question QALD_010

D. START ANSWERS

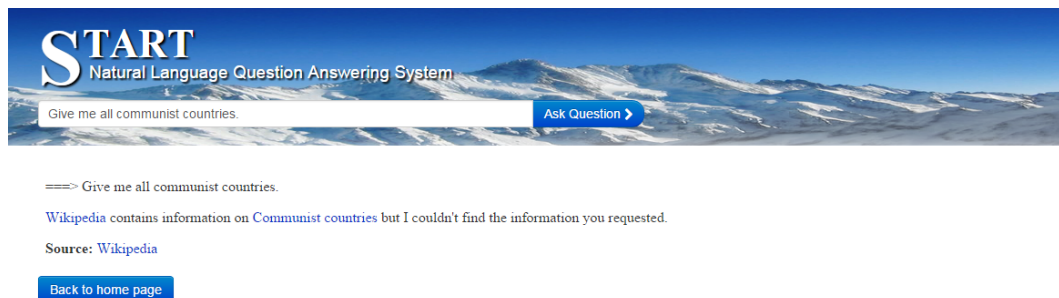


Figure D.2: START QA system answering the question QALD_028

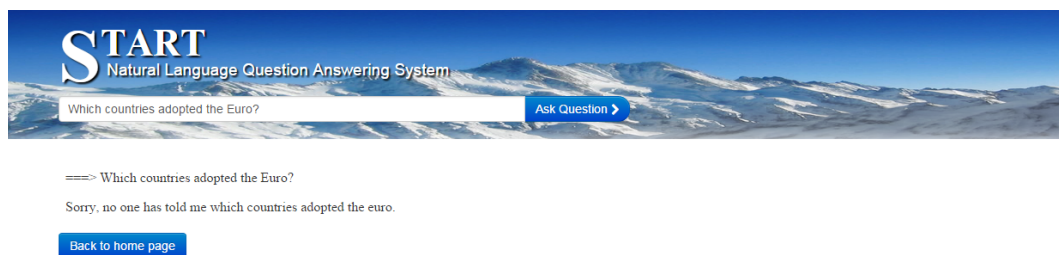


Figure D.3: START QA system answering the question QALD_032

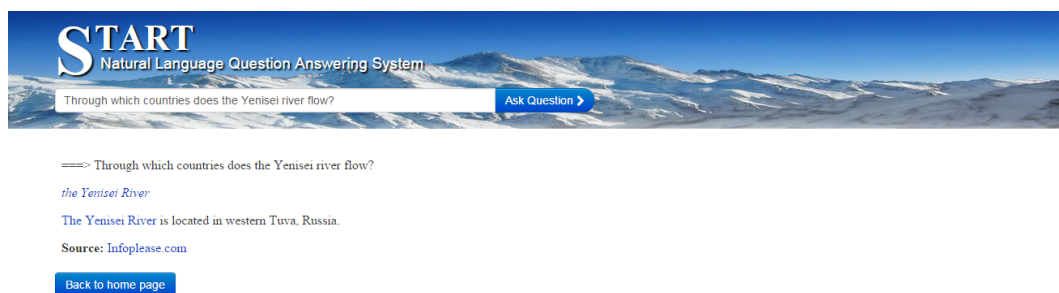
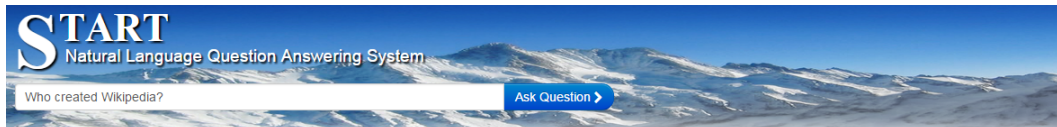


Figure D.4: START QA system answering the question QALD_036



==> Who created Wikipedia?

[Wikipedia](#)

Wikipedia is a free, open content online encyclopedia created through the collaborative effort of a community of users known as *Wikipedians*. Anyone registered on the site can create an article for publication; registration is not required to edit articles. The site's name comes from [wiki](#), a server program that enables anyone to edit Web site content through their Web browser.

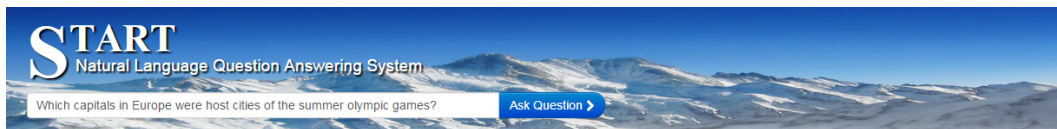
Jimmy Wales and Larry Sanger co-founded Wikipedia as an offshoot of an earlier encyclopedia project, Nupedia, in January 2001. Originally, Wikipedia was created to provide content for Nupedia. However, as the wiki site became established it soon grew beyond the scope of the earlier project. As of January 2008, the encyclopedia offered over four million articles, 2,175,080 of which were in English. At that same time, Alexa ranked Wikipedia as the eighth-most popular site on the Internet. Wikipedia was the only non-commercial site of the top ten.

Criticisms of Wikipedia include assertions that its openness makes it unreliable and unauthoritative. Because articles don't include bylines, authors aren't publicly accountable for what they write. Similarly, because anyone can edit any article, the site's entries are vulnerable to unscrupulous edits. In August 2007, Virgil Griffiths created a site, [WikiScanner](#), where users could track the sources of edits to Wikipedia entries. Griffiths reported that self-serving edits typically involved whitewashing or removal of criticism of a person or organization or, conversely, insertion of negative comments into the entry about a competitor. Wikipedia depends upon the vigilance of editors to find and reverse such changes to content.

In addition to the encyclopedia, the non-profit Wikipedia foundation oversees several other open-content projects, including:

- Wiktionary, a dictionary and thesaurus
- Wikibooks, a collection of free texts and other books
- Wikiquote, a collection of quotations
- Wikisource, a collection of free source documents
- Wikiversity, a collection of free learning materials
- Wikispecies, a directory of species
- Meta-Wiki, which coordinates all the other projects.

Figure D.5: START QA system answering the question QALD_062



==> Which capitals in Europe were host cities of the summer olympic games?

I don't know the answer to this question. Sorry.

[Back to home page](#)

Figure D.6: START QA system answering the question QALD_074

D. START ANSWERS

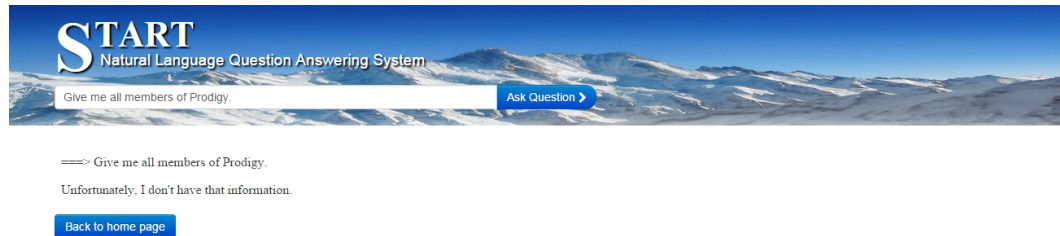


Figure D.7: START QA system answering the question QALD_114

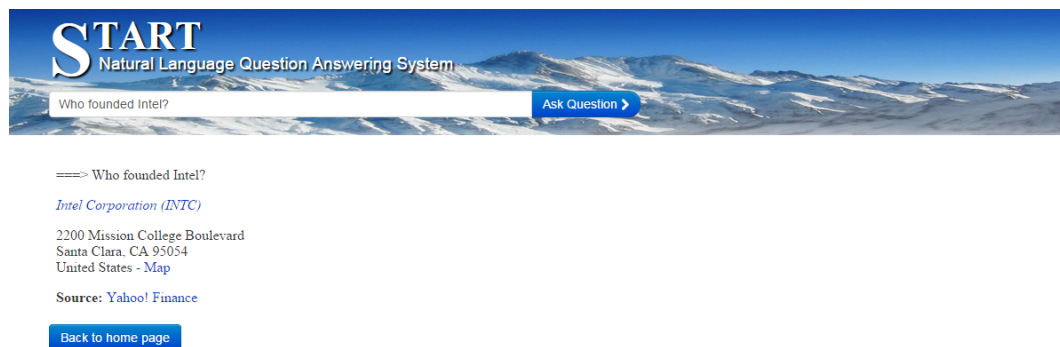


Figure D.8: START QA system answering the question QALD_141

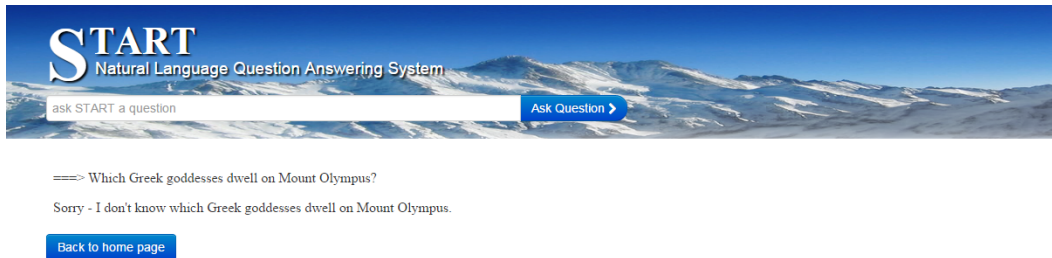


Figure D.9: START QA system answering the question QALD_155

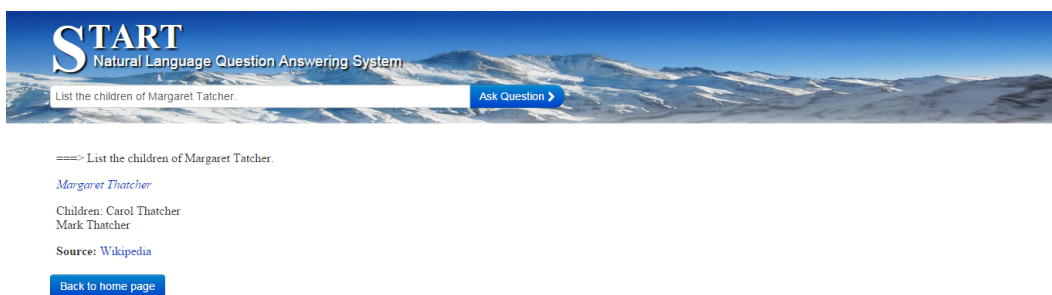


Figure D.10: START QA system answering the question QALD_176

E

Wolfram Alpha Answers

Below we show the output of the Wolfram Alpha QA system to the question dataset used in the experiments described in Chapter 5 - Evaluation.

The screenshot shows the Wolfram Alpha interface. At the top, the Wolfram Alpha logo is displayed with the tagline 'computational knowledge engine'. Below the logo is a search bar containing the question 'In which country does the Nile start?'. To the right of the search bar are icons for saving, sharing, and a star. Below the search bar are links for 'Examples' and 'Random'. The main content area is divided into several sections: 'Input interpretation' showing 'Nile (music act)' and 'country'; 'Result' showing 'United States'; 'Basic information' showing a table with 'name', 'active dates', and 'country'; 'Name' showing a table with 'full name', 'alternate names', and 'internet code'; 'Flag' showing the flag of the United States; and 'Location' showing a map of the United States. A 'Show mesh' button is located at the bottom right of the location section.

WolframAlpha computational knowledge engine

In which country does the Nile start?

Examples Random

Input interpretation:

Nile (music act) country

Result:

United States

Basic information:

name	Nile
active dates	1993 to present
country	United States

Name:

full name	United States of America
alternate names	America US USA U.S. U.S.A.
internet code	.us

More

Flag:

Location:

Show mesh

Figure E.1: Wolfram Alpha QA system answering the question QALD_010

E. WOLFRAM ALPHA ANSWERS

WolframAlpha computational... knowledge engine

Give me all communist countries.

Examples Random

Using closest Wolfram|Alpha interpretation: **Give communist**

More interpretations: [Give all](#)

Input interpretation:
communist (English word)

Definitions: [Show examples](#)

1	noun	a socialist who advocates communism
2	adjective	relating to or marked by communism

American pronunciation:
k'omyuhnuhst (IPA: k' ɒmjənəst)

Hyphenation:
com-mun-ist (9 letters | 3 syllables)

Overall typical frequency:

written:	2381 st most common	(1 in 22 727 words)	(> 99% adjective 0% noun)
spoken:	2275 th most common	(1 in 45 455 words)	

[Definition »](#)

(includes some inflected forms)

Figure E.2: Wolfram Alpha QA system answering the question QALD_028

 **WolframAlpha** computational... knowledge engine

Which countries adopted the Euro?☆⌵

   Examples Random

Using closest Wolfram|Alpha interpretation: **countries Euro**?

More interpretations: **countries Euro**


Input interpretation:

1 euro coin

countries of circulation

Result

Vatican City | Germany | France | Italy | Spain | Netherlands | Belgium | Austria | Greece | Finland | Ireland | Portugal | Slovakia | Slovenia | Luxembourg | Cyprus | Estonia | Réunion | Malta | Monaco | Martinique | Montenegro | Kosovo | Andorra | Guadeloupe | San Marino | French Guiana | Mayotte | Saint Pierre and Miquelon


 Sources  Download page POWERED BY THE WOLFRAM LANGUAGE

 Standard computation time exceeded... Try again with additional computation time »


Give us your feedback: Send

Figure E.3: Wolfram Alpha QA system answering the question QALD_032

E. WOLFRAM ALPHA ANSWERS

 **WolframAlpha** computational knowledge engine

Through which countries does the Yenisei river flow? ☆ =

[Examples](#) [Random](#)

Input interpretation:

Yenisei countries

Result:

Russia | Mongolia

Names: More

	Russia	Mongolia
full name	Russian Federation	Mongolia
full native name	Rossiyskaya Federatsiya	Mongol Uls
internet code	.ru	.mn

Flags:

Russia








Mongolia



Figure E.4: Wolfram Alpha QA system answering the question QALD_036

 **WolframAlpha** computational...
knowledge engine

    Examples Random

Using closest Wolfram|Alpha interpretation: **created Wikipedia** 

Input interpretation:

Result:

13/01/2001

Date formats: More formats/calendars

13/01/2001 (day/month/year)

Time difference from today (Friday, December 12, 2014):

13 years 10 months 30 days ago

725 weeks 6 days ago

5081 days ago

13.91 years ago

Time in 2001: More

13th day

2nd week

Figure E.5: Wolfram Alpha QA system answering the question QALD_062

E. WOLFRAM ALPHA ANSWERS



Which capitals in Europe were host cities of the summer olympic games? ☆

Using closest Wolfram|Alpha interpretation: **host cities of the summer olympic games** ?

More interpretations: **capitals**

Input interpretation:

Olympic Games	summer	host city
---------------	--------	-----------

Result

(data not available)

Figure E.6: Wolfram Alpha QA system answering the question QALD_074

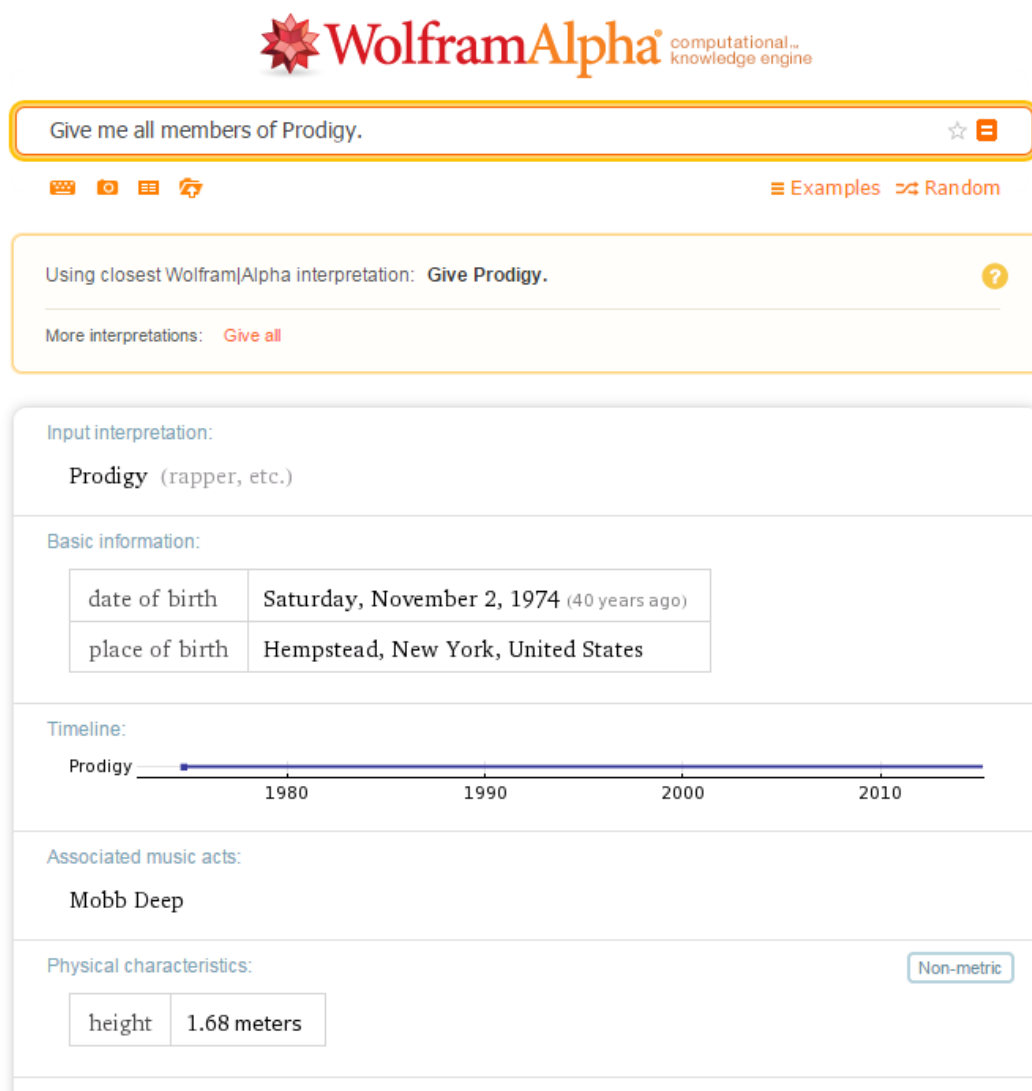


Figure E.7: Wolfram Alpha QA system answering the question QALD_114

E. WOLFRAM ALPHA ANSWERS

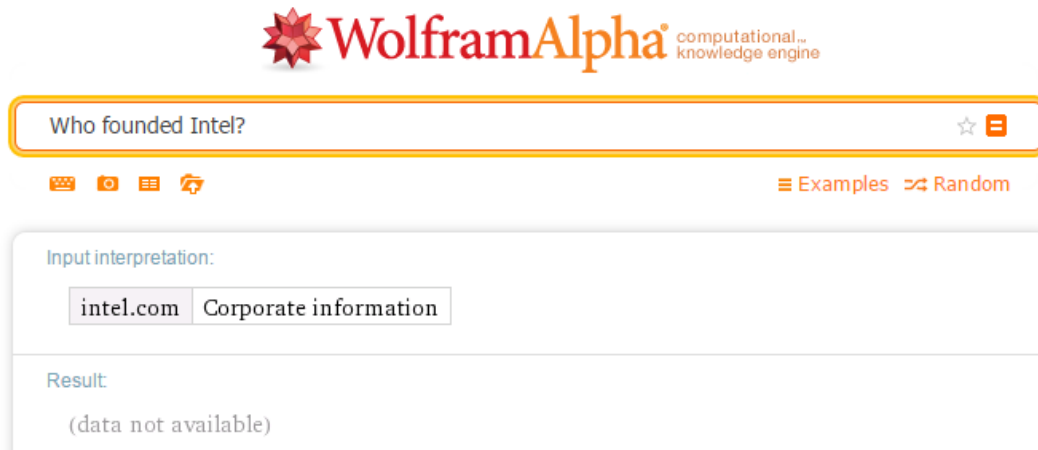



Figure E.8: Wolfram Alpha QA system answering the question QALD_141







Figure E.9: Wolfram Alpha QA system answering the question QALD_155

 computational knowledge engine

List the children of Margaret Thatcher.

☆



Examples ↗ Random

Using closest Wolfram|Alpha interpretation: the children of Margaret Thatcher?

Input interpretation:

Margaret Thatcher children

Result

Carol Thatcher | Mark Thatcher

Familial relationships:

Show full dates

Parents:

Alfred Roberts | Beatrice Roberts

Sibling:

Muriel Roberts

Spouse:

Denis Thatcher (1951–2003)

Children:

Carol Thatcher | Mark Thatcher

Figure E.10: Wolfram Alpha QA system answering the question QALD_176

References

- AHN, DAVID, STEVEN SCHOCKAERT, MARTINE DE COCK AND ETIENNE KERRE, 2006. Supporting Temporal Question Answering: Strategies for Offline Data Collection. In *Proceedings of the Fifth International Workshop on Inference in Computational Semantics (ICoS-5)*. [Cited at pg. [39](#), [41](#), [42](#), [73](#), [79](#)]
- AHN, KISUH, JOHAN BOS, DAVID KOR, MALVINA NISSIM, BONNIE L. WEBBER AND JAMES R. CURRAN, 2005. Question Answering with QED at TREC 2005. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, November 15-18, 2005*, volume Special Publication 500-266. National Institute of Standards and Technology (NIST). [Cited at pg. [30](#)]
- ALLEN, JAMES F., 1983. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26(11):832–843. ISSN 0001-0782. [Cited at pg. [xvii](#), [74](#), [75](#)]
- AMARAL, CARLOS, ADÁN CASSAN, HELENA FIGUEIRA, ANDRÉ MARTINS, AFONSO MENDES, PEDRO MENDES, JOSÉ PINA AND CLÁUDIA PINTO, 2009. Priberam’s Question Answering System in QA@CLEF 2008. In *Proceedings of the 9th Cross-language evaluation forum conference on Evaluating systems for multilingual and multimodal information access*, CLEF’08, pages 337–344. Springer-Verlag, Berlin, Heidelberg. ISBN 3-642-04446-8, 978-3-642-04446-5. [Cited at pg. [25](#), [26](#), [27](#)]
- AMARAL, CARLOS, HELENA FIGUEIRA, ANDRÉ MARTINS, AFONSO MENDES, PEDRO MENDES AND CLÁUDIA PINTO, 2006. Priberam’s Question Answering System for Portuguese. In *Proceedings of the 6th international conference on Cross-Language Evaluation Forum: accessing Multilingual Information Repositories*, CLEF’05, pages 410–419. Springer-Verlag, Berlin, Heidelberg. ISBN 3-540-45697-X, 978-3-540-45697-1. [Cited at pg. [25](#)]

REFERENCES

- BANKO, MICHELE, ERIC BRILL, SUSAN DUMAIS AND JIMMY LIN, 2002. AskMSR: Question Answering Using the Worldwide Web. In *Proceedings of 2002 AAAI Spring Symposium on Mining Answers from Texts and Knowledge Bases*, pages 7–9. [Cited at pg. [34](#), [37](#)]
- BRANCO, ANTÓNIO, LINO RODRIGUES, JOÃO SILVA AND SARA SILVEIRA, 2008a. Real-Time Open-Domain QA on the Portuguese Web. In *Proceedings of the 11th Ibero-American conference on AI (IBERAMIA '08)*, pages 322–331. Springer-Verlag, Berlin, Heidelberg. ISBN 978-3-540-88308-1. [Cited at pg. [26](#), [27](#)]
- BRANCO, ANTÓNIO, LINO RODRIGUES, JOÃO SILVA AND SARA SILVEIRA, 2008b. XisQuê: An Online QA Service for Portuguese. In *Proceedings of the 8th International Conference on Computational Processing of the Portuguese Language (PROPOR'08)*, pages 232–235. Springer-Verlag, Berlin, Heidelberg. ISBN 978-3-540-85979-6. [Cited at pg. [26](#)]
- BRANCO, ANTÓNIO AND JOÃO SILVA, 2006. A suite of shallow processing tools for Portuguese: LX-Suite. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics*, EACL '06, pages 179–182. Association for Computational Linguistics, Stroudsburg, PA, USA. [Cited at pg. [53](#)]
- CARDOSO, NUNO, DAVID BATISTA, FRANCISCO J. LÓPEZ-PELLICER AND MÁRIO J. SILVA, 2009. Where in the Wikipedia Is That Answer? The XLDB at the GikiCLEF 2009 Task. In CAROL PETERS, GIORGIO MARIA DI NUNZIO, MIKKO KURIMO, DJAMEL MOSTEFA, ANSELMO PEÑAS AND GIOVANNA RODA, editors, *CLEF*, volume 6241 of *Lecture Notes in Computer Science*, pages 305–309. Springer. ISBN 978-3-642-15753-0. [Cited at pg. [30](#), [31](#), [32](#)]
- CARDOSO, NUNO, IUSTIN DORNESCU, SVEN HARTRUMPF AND JOHANNES LEVELING, 2010. Revamping Question Answering with a Semantic Approach over World Knowledge. In MARTIN BRASCHLER, DONNA HARMAN AND EMANUELE PIANTA, editors, *CLEF 2010 LABs and Workshops, Notebook Papers, 22-23 September 2010, Padua, Italy*. ISBN 978-88-904810-0-0. [Cited at pg. [30](#)]
- CARVALHO, GRACINDA, DAVID DE MATOS AND VITOR ROCIO, 2009. IdSay: Question Answering for Portuguese. In CAROL PETERS, THOMAS DESELAERS, NICOLA FERRO,

- JULIO GONZALO, GARETH JONES, MIKKO KURIMO, THOMAS MANDL, ANSELMO PEÑAS AND VIVIEN PETRAS, editors, *Evaluating Systems for Multilingual and Multimodal Information Access*, volume 5706 of *Lecture Notes in Computer Science*, pages 345–352. Springer Berlin - Heidelberg. ISBN 978-3-642-04446-5. [Cited at pg. [25](#), [26](#), [27](#)]
- CARVALHO, GRACINDA, DAVID DE MATOS AND VITOR ROCIO, 2010. Improving Id-Say: A Characterization of Strengths and Weaknesses in Question Answering Systems for Portuguese. In THIAGO PARDO, ANTÓNIO BRANCO, ALDEBARO KLAUTAU, RENATA VIEIRA AND VERA DE LIMA, editors, *Computational Processing of the Portuguese Language*, volume 6001 of *Lecture Notes in Computer Science*, pages 1–10. Springer Berlin / Heidelberg. ISBN 978-3-642-12319-1. [Cited at pg. [25](#), [133](#)]
- CASSAN, ADÁN, HELENA FIGUEIRA, ANDRÉ F. T. MARTINS, AFONSO MENDES, PEDRO MENDES, CLÁUDIA PINTO AND DANIEL VIDAL, 2006. Priberam’s Question Answering System in a Cross-Language Environment. In CAROL PETERS, PAUL CLOUGH, FREDRIC C. GEY, JUSSI KARLGREN, BERNARDO MAGNINI, DOUGLAS W. OARD, MAARTEN DE RIJKE AND MAXIMILIAN STEMPFHUBER, editors, *Proceedings of the 6th international conference on Cross-Language Evaluation Forum: accessing Multilingual Information Repositories*, volume 4730 of *Lecture Notes in Computer Science*, pages 300–309. Springer. ISBN 978-3-540-74998-1. [Cited at pg. [25](#)]
- CLARKE, CHARLES L. A., GORDON V. CORMACK AND THOMAS R. LYNAM, 2001. Exploiting Redundancy in Question Answering. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’01, pages 358–365. ACM, New York, NY, USA. ISBN 1-58113-331-6. [Cited at pg. [34](#), [37](#)]
- COHEUR, LUÍSA, ANA MENDES, JOÃO GUIMARÃES, NUNO J. MAMEDE AND RICARDO RIBEIRO, 2009. Question interpretation in QA@L2F. In *Proceedings of the 9th Cross-language evaluation forum conference on Evaluating systems for multilingual and multimodal information access*, CLEF’08, pages 377–384. Springer-Verlag, Berlin, Heidelberg. ISBN 3-642-04446-8, 978-3-642-04446-5. [Cited at pg. [25](#), [26](#), [27](#)]

REFERENCES

- COSTA, FRANCISCO, 2004. Verbal Conjugation in Portuguese. Internal report, University of Lisbon, Department of Informatics. [Cited at pg. [52](#)]
- COSTA, FRANCISCO AND ANTÓNIO BRANCO, 2013. Full-fledged temporal processing: bridging the gap between deep linguistic processing and temporal extraction. *Journal of Language Modelling*, 1(1):97–154. [Cited at pg. [134](#)]
- COSTA, LUÍS FERNANDO, 2005. Esfinge - Resposta a perguntas usando a Rede. In *Proceedings of IADIS Ibero-Americana*, pages 616–619. IADIS Press. [Cited at pg. [24](#), [26](#)]
- COSTA, LUÍS FERNANDO, 2006. Esfinge: a Question Answering System in the Web using the Web. In *Proceedings of the 11 th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. The Association for Computer Linguistics. ISBN 1-932432-59-0. [Cited at pg. [24](#), [27](#)]
- CUCERZAN, SILVIU AND EUGENE AGICHTEIN, 2005. Factoid Question Answering over Unstructured and Structured Web Content. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *Proceedings of the Text REtrieval Conference (TREC)*, volume Special Publication 500-266. National Institute of Standards and Technology (NIST). [Cited at pg. [9](#)]
- DALMAS, TIPHAIN AND BONNIE L. WEBBER, 2007. Answer Comparison in Automated Question Answering. *Journal of Applied Logic*, 5(1):104–120. [Cited at pg. [30](#)]
- DORNESCU, IUSTIN, 2009. Semantic QA for Encyclopaedic Questions: EQUAL in Giki-CLEF. In CAROL PETERS, GIORGIO MARIA DI NUNZIO, MIKKO KURIMO, DJAMEL MOSTEFA, ANSELMO PEÑAS AND GIOVANNA RODA, editors, *CLEF*, volume 6241 of *Lecture Notes in Computer Science*, pages 326–333. Springer. ISBN 978-3-642-15753-0. [Cited at pg. [31](#), [32](#)]
- DUMAIS, SUSAN, MICHELE BANKO, ERIC BRILL, JIMMY LIN AND ANDREW NG, 2002. Web Question Answering: Is More Always Better? In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 291–298. ACM, New York, NY, USA. ISBN 1-58113-561-0. [Cited at pg. [34](#), [102](#)]

REFERENCES

- FERREIRA, EDUARDO, JOÃO Balsa AND ANTÓNIO BRANCO, 2007. Combining Rule-Based and Statistical Models for Named Entity Recognition of Portuguese. In *Proceedings of Workshop em Tecnologia da Informação e de Linguagem Natural*, pages 1615–1624. [Cited at pg. [53](#)]
- FREITAS, CLÁUDIA, 2012. A lusofonia na Wikipédia em 150 tópicos. *Linguamatica*, 4(1):9–18. ISSN 1647. [Cited at pg. [98](#)]
- GAIZAUSKAS, ROBERT J., MARK A. GREENWOOD, HENK HARKEMA, MARK HEPPLE, HORACIO SAGGION AND ATHEESH SANKA, 2005. The University of Sheffield's TREC 2005 Q&A Experiments. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, November 15-18, 2005*, volume Special Publication 500-266. National Institute of Standards and Technology (NIST). [Cited at pg. [27](#), [32](#)]
- GONÇALVES, PATRICIA AND ANTONIO BRANCO, 2014a. Answering List Questions using Web as a corpus. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 81–84. Association for Computational Linguistics, Gothenburg, Sweden. [Cited at pg. [129](#)]
- GONÇALVES, PATRICIA NUNES AND ANTÓNIO BRANCO, 2014b. Open-Domain Web-Based List Question Answering with LX-ListQuestion. In *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics, WIMS '14*, pages 43:1–43:6. ACM, New York, NY, USA. ISBN 978-1-4503-2538-7. [Cited at pg. [131](#)]
- GREEN, BERT F., JR., ALICE K. WOLF, CAROL CHOMSKY AND KENNETH LAUGHERY, 1961. Baseball: an Automatic Question-Answerer. In *Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference*, IRE-AIEE-ACM '61 (Western), pages 219–224. ACM, New York, NY, USA. [Cited at pg. [3](#)]
- GUDA, VANITHA, SURESH SANAMPUDI AND LAKSHMI MANIKYAMBA, 2011. Approaches for Question Answering Systems. *International Journal of Engineering Science and Technology (IJEST)*, 3(2):990–995. ISSN 0975-5462. [Cited at pg. [9](#)]
- HARABAGIU, SANDA AND COSMIN ADRIAN BEJAN, 2005. Question Answering Based on Temporal Inference. In *Proceedings of the AAAI-2005 Workshop on Inference for Textual Question Answering*. [Cited at pg. [12](#), [40](#), [42](#), [76](#), [78](#), [79](#), [80](#)]

REFERENCES

- HARTRUMPF, SVEN, 2008. Semantic Decomposition for Question Answering. In MALIK GHALLAB, CONSTANTINE D. SPYROPOULOS, NIKOS FAKOTAKIS AND NIKOLAOS M. AVOURIS, editors, *ECAI*, volume 178 of *Frontiers in Artificial Intelligence and Applications*, pages 313–317. IOS Press. ISBN 978-1-58603-891-5. [Cited at pg. [31](#)]
- HARTRUMPF, SVEN AND JOHANNES LEVELING, 2010. GIRSA-WP at GikiCLEF: Integration of structured information and decomposition of questions. In *10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009, Corfu, Greece, September 30-October 2, Revised Selected Papers*, Lecture Notes in Computer Science (LNCS). Springer. (to appear). [Cited at pg. [31](#), [32](#)]
- HICKL, ANDREW, KIRK ROBERTS, BRYAN RINK, JEREMY BENSLEY, TOBIAS JUNGEN, YING SHI AND JOHN WILLIAMS, 2007. Question Answering with LCC’s Chaucer-2 at TREC 2007. In *Proceedings of the Sixteenth Text REtrieval Conference*. [Cited at pg. [28](#)]
- HICKL, ANDREW, JOHN WILLIAMS, JEREMY BENSLEY, KIRK ROBERTS, YING SHI AND BRYAN RINK, 2006. Question Answering with LCC’s CHAUCER at TREC 2006. In *TREC*. [Cited at pg. [28](#), [32](#)]
- KAISSER, MICHAEL AND TILMAN BECKER, 2004. Question Answering by Searching Large Corpora With Linguistic Methods. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *TREC*, volume Special Publication 500-261. National Institute of Standards and Technology (NIST). [Cited at pg. [9](#), [35](#), [37](#), [38](#)]
- KATZ, BORIS, 1997. Annotating the World Wide Web using Natural Language. In LUC DEVROYE AND CLAUDE CHRISMENT, editors, *Computer-Assisted Information Retrieval - RIAO 1997, 5th International Conference, McGill University, Montreal, Canada, June 25-27, 1997*, pages 136–159. [Cited at pg. [33](#), [37](#), [111](#)]
- KATZ, BORIS, JIMMY J. LIN, DANIEL LORETO, WESLEY HILDEBRANDT, MATTHEW W. BILOTTI, SUE FELSHIN, AARON FERNANDES, GREGORY MARTON AND FEDERICO MORA, 2003. Integrating Web-based and Corpus-based Techniques for Question Answering. In *Proceedings of the 14th Annual Text Retrieval Conference*, pages 426–435. [Cited at pg. [33](#)]

REFERENCES

- KATZ, BORIS, GREGORY MARTON, GARY BORCHARDT, ALEXIS BROWNELL, SUE FELSHIN, DANIEL LORETO, JESSE LOUIS-ROSENBERG, BEN LU, FEDERICO MORA, STEPHAN STILLER, ÖZLEM UZUNER AND ANGELA WILCOX, 2005. External Knowledge Sources for Question Answering. In *Proceedings of the 14th Annual Text Retrieval Conference (TREC'2005)*. [Cited at pg. 37, 38]
- KOR, KIAN WEI, 2005. *Improving Answer Precision And Recall of List Questions*. Master's thesis, University of Edinburgh, School of Informatics. [Cited at pg. 30, 32]
- KUPIEC, JULIAN, 1993. MURAX: A Robust Linguistic Approach for Question Answering Using an On-Line Encyclopedia. In *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Pittsburgh, PA, USA, June 27 - July 1, 1993, pages 181–190. [Cited at pg. 33, 37]
- KWOK, CODY, OREN ETZIONI AND DANIEL S. WELD, 2001. Scaling Question Answering to the Web. *ACM Transactions on Information Systems (TOIS)*, 19(3):242–262. ISSN 1046-8188. [Cited at pg. 9, 34, 37]
- LEHNERT, WENDY G., 1978. *The Process of Question Answering: A Computer Simulation of Cognition*. Lawrence Erlbaum Associates, Hillsdale, NJ. Based on 1977 Yale PhD thesis. [Cited at pg. 3]
- LLORET, ELENA, HECTOR LLORENS, PALOMA MOREDA, ESTELA SAQUETE AND MANUEL PALOMAR, 2011. Text Summarization Contribution to Semantic Question Answering: New Approaches for Finding Answers on the Web. *Journal International Journal of Intelligent Systems*, 26(12):1125–1152. ISSN 0884-8173. [Cited at pg. 36, 37]
- MAGNINI, BERNARDO, MATTEO NEGRI, ROBERTO PREVETE AND HRISTO TANEV, 2001. Multilingual Question/Answering: the DIOGENE System. In *Proceedings of the 10th Text Retrieval Conference*. [Cited at pg. 35, 37]
- MAZIERO, ERICK G., THIAGO A. S. PARDO, ARIANI DI FELIPPO AND BENTO C. DIAS-DA SILVA, 2008. A Base De Dados Lexical e a Interface Web Do TeP 2.0: Thesaurus Eletrônico Para O Português Do Brasil. In *Proceedings of the XIV Brazilian Symposium on Multimedia and the Web*, WebMedia '08, pages 390–392. ACM, New York, NY, USA. ISBN 978-85-7669-199-0. [Cited at pg. 53]

REFERENCES

- MOLDOVAN, DAN, CHRISTINE CLARK AND SANDA HARABAGIU, 2005. Temporal Context Representation and Reasoning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. [Cited at pg. [41](#), [42](#), [83](#)]
- MOLDOVAN, DAN, SANDA HARABAGIU, MARIUS PAȘCA, RADA MIHALCEA, ROXANA GIRJU, RICHARD GOODRUM AND VASILE RUS, 2000. The Structure and Performance of an Open-domain Question Answering System. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 563–570. Association for Computational Linguistics, Stroudsburg, PA, USA. [Cited at pg. [58](#)]
- MOTA, CRISTINA, 2012. Resultados págicos: participação, medidas e pontuação. *Linguistica*, 4(1):77–91. ISSN 1647. [Cited at pg. [26](#)]
- PAȘCA, MARIUS, 2003. *Open-Domain Question Answering from Large Text Collections*. CSLI Studies in Computational Linguistics. CSLI, Stanford, California. [Cited at pg. [2](#), [54](#)]
- PAȘCA, MARIUS, 2008. Towards Temporal Web Search. In *Proceedings of the 2008 ACM Symposium on Applied Computing*, SAC '08, pages 1117–1121. ACM, New York, NY, USA. ISBN 978-1-59593-753-7. [Cited at pg. [11](#), [39](#), [42](#), [80](#)]
- PINTO, DAVID, MICHAEL BRANSTEIN, RYAN COLEMAN, W. BRUCE CROFT, MATTHEW KING, WEI LI AND XING WEI, 2002. QuASM: A System for Question Answering Using Semi-Structured Data. In *Proceedings of the Joint Conference on Digital Libraries (JCDL) 2002*, pages 46–55. [Cited at pg. [35](#), [37](#)]
- PUSTEJOVSKY, J., J. WIEBE AND M. MAYBURY, 2002. Multi-Perspective and Temporal Question Answering. In *Proceedings of the conference on International Language Resources and Evaluation - Workshop on Question Answering: Strategy and Resources*. [Cited at pg. [79](#)]
- PUSTEJOVSKY, JAMES, JOSÉ CASTAÑO, ROBERT INGRIA, ROSER SAURÍ, ROBERT GAIZAUSKAS, ANDREA SETZER AND GRAHAM KATZ, 2003. TimeML: Robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*. [Cited at pg. [75](#)]

REFERENCES

- RADEV, DRAGOMIR, WEIGUO FAN, HONG QI, HARRIS WU AND AMARDEEP GREWAL, 2002. Probabilistic Question Answering on the Web. In *Proceedings of the 11th International Conference on World Wide Web*, WWW '02, pages 408–419. ACM, New York, NY, USA. ISBN 1-58113-449-5. [Cited at pg. [9](#), [35](#), [37](#), [39](#)]
- RADEV, DRAGOMIR AND BETH SUNDHEIM, 2002. Using TimeML in Question Answering. [Cited at pg. [38](#), [42](#), [76](#), [79](#), [82](#)]
- RAZMARA, MAJID AND LEILA KOSSEIM, 2008. Answering List Questions using Co-occurrence and Clustering. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*. European Language Resources Association. [Cited at pg. [29](#), [32](#)]
- RODRIGUES, LINO MIGUEL SILVA, 2007. *Infra-Estruturas de um serviço online de resposta-a-perguntas com base na web portuguesa*. Master's thesis, Universidade de Lisboa, Faculdade de Ciências, Departamento de Informática. [Cited at pg. [26](#)]
- RODRIGUES, RICARDO AND HUGO OLIVEIRA, 2012. Uma abordagem ao Páxico baseada no Processamento e Análise de Sintagmas dos Tópicos. *Linguamatica*, 4(1):31–39. ISSN 1647. [Cited at pg. [26](#), [106](#)]
- SAIAS, JOSÉ AND PAULO QUARESMA, 2009. The Senso Question Answering System at QA@CLEF 2008. In *Proceedings of the 9th Cross-language evaluation forum conference on Evaluating systems for multilingual and multimodal information access*, CLEF'08, pages 337–344. Springer-Verlag, Berlin, Heidelberg. ISBN 3-642-04446-8, 978-3-642-04446-5. [Cited at pg. [24](#), [26](#), [27](#)]
- SAQUETE, ESTELA, JOSÉ LUIS VICEDO GONZÁLEZ, PATRICIO MARTÍNEZ-BARCO, RAFAEL MUÑOZ AND HECTOR LLORENS, 2009. Enhancing QA Systems with Complex Temporal Question Processing Capabilities. *Journal of Artificial Intelligence Research (JAIR)*, 35:775–811. [Cited at pg. [41](#), [42](#), [78](#), [79](#), [80](#), [82](#)]
- SARMENTO, LUÍS, 2006. Hunting Answers with RAPOSA (FOX). In JOSÉ LUÍS VICEDO ALESSANDRO NARDI, CAROL PETERS, editor, *Cross Language Evaluation Forum: Working Notes for the CLEF 2006 Workshop*. Springer. [Cited at pg. [25](#)]

REFERENCES

- SARMENTO, LUÍS, JORGE TEIXEIRA AND C. OLIVEIRA, 2008a. Experiments with Query Expansion in the Raposa (Fox) Question Answering System. In CAROL PETERS AND F. BORRI, editors, *Cross Language Evaluation Forum: Working Notes for the CLEF 2008 Workshop*. Springer. [Cited at pg. [25](#), [26](#)]
- SARMENTO, LUÍS, JORGE TEIXEIRA AND EUGÉNIO C. OLIVEIRA, 2008b. Assessing the Impact of Thesaurus-Based Expansion Techniques in QA-Centric IR. In CAROL PETERS, THOMAS DESELAERS, NICOLA FERRO, JULIO GONZALO, GARETH J. F. JONES, MIKKO KURIMO, THOMAS MANDL, ANSELMO PEÑAS AND VIVIEN PETRAS, editors, *Cross Language Evaluation Forum: Working Notes for the CLEF*, volume 5706 of *Lecture Notes in Computer Science*, pages 325–332. Springer. ISBN 978-3-642-04446-5. [Cited at pg. [27](#)]
- SAURÍ, ROSER, JESSICA LITTMAN, ROBERT GAIZAUSKAS, ANDREA SETZER AND JAMES PUSTEJOVSKY, 2006. TimeML Annotation Guidelines, Version 1.2.1. [Cited at pg. [75](#)]
- SCHILDER, FRANK AND CHRISTOPHER HABEL, 2003. Temporal Information Extraction for Temporal Question Answering. In MARK T. MAYBURY, editor, *New Directions in Question Answering*, pages 35–44. AAAI Press. ISBN 1-57735-184-3. [Cited at pg. [11](#), [39](#), [42](#), [79](#), [134](#)]
- SCHOCKAERT, STEVEN, DAVID AHN, MARTINE DE COCK AND ETIENNE E. KERRE, 2006. Question answering with imperfect temporal information. In *Proceedings of the 7th International Conference on Flexible Query Answering Systems, LNAI 4027*, pages 647–658. [Cited at pg. [41](#), [42](#), [80](#), [83](#), [84](#), [133](#)]
- STRZALKOWSKI, TOMEK AND SANDA HARABAGIU, 2007. *Advances in Open Domain Question Answering*. Springer Publishing Company, Incorporated, 1st edition. ISBN 1402047452, 9781402047459. [Cited at pg. [3](#)]
- TAO, CUI, HAROLD SOLBRIG, DEEPAK SHARMA, WEI-QI WEI, GUERGANA SAVOVA AND CHRISTOPHER CHUTE, 2010. Time-Oriented Question Answering from Clinical Narratives Using Semantic-Web Techniques. In PETERF. PATEL-SCHNEIDER, YUE PAN, PASCAL HITZLER, PETER MIKA, LEI ZHANG, JEFFZ. PAN, IAN HORROCKS AND BIRTE GLIMM, editors, *The Semantic Web – ISWC 2010*, volume 6497 of *Lecture Notes in*

REFERENCES

- Computer Science*, pages 241–256. Springer Berlin Heidelberg. ISBN 978-3-642-17748-4. [Cited at pg. [12](#), [40](#), [42](#), [80](#)]
- TEUFEL, SIMONE, 2007. An Overview of Evaluation Methods in TREC Ad Hoc Information Retrieval and TREC Question Answering. In LAILA DYBKJÆR, HOLMER HEMSEN AND WOLFGANG MINKER, editors, *Evaluation of Text and Speech Systems*, volume 37 of *Text, Speech and Language Technology*, pages 163–186. Springer Netherlands. ISBN 978-1-4020-5816-5. [Cited at pg. [21](#)]
- UZZAMAN, NAUSHAD, HECTOR LLORENS AND JAMES F. ALLEN, 2012. Evaluating Temporal Information Understanding with Temporal Question Answering. In *Proceedings of the Sixth International Conference of Semantic Computing (ICSC-2012)*, pages 79–82. IEEE Computer Society. ISBN 978-1-4673-4433-3. [Cited at pg. [73](#), [74](#)]
- WANG, RICHARD C., NICO SCHLAEFER, WILLIAM W. COHEN AND ERIC NYBERG, 2008. Automatic Set Expansion for List Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (2008)*, pages 947–954. ACL. [Cited at pg. [29](#)]
- WEBBER, BONNIE, CLAIRE GARDENT AND JOHAN BOS, 2002. Position statement: Inference in Question Answering. In *Proceedings of the LREC Workshop on Question Answering: Strategy and Resources, Las Palmas, Gran Canaria, Spain*. [Cited at pg. [29](#)]
- WHITTAKER, EDWARD W. D., JOSEF R. NOVAK, PIERRE CHATAIN AND SADAOKI FURUI, 2006. TREC 2006 Question Answering Experiments at Tokyo Institute of Technology. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *Proceedings of the Fifteenth Text REtrieval Conference, TREC 2006, Gaithersburg, Maryland, November 14-17, 2006*, volume Special Publication 500-272. National Institute of Standards and Technology (NIST). [Cited at pg. [28](#), [32](#)]
- WOODS, W., R. KAPLAN AND B. NASH-WEBBER, 1974. The Lunar Sciences Natural Language Information System. Final Report 2378, Bolt, Beranek and Newman, Inc., Cambridge, MA. [Cited at pg. [3](#)]
- WU, MIN AND TOMEK STRZALKOWSKI, 2006. Utilizing Co-Occurrence of Answers in Question Answering. In *ACL 2006, 21st International Conference on Computational*

REFERENCES

- Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. The Association for Computer Linguistics. [Cited at pg. [27](#), [32](#)]
- WU, MINJI AND AMÉLIE MARIAN, 2011. A Framework for Corroborating Answers from Multiple Web Sources. *Information Systems*, 36(2):431–449. ISSN 0306-4379. [Cited at pg. [36](#), [37](#)]
- YAHYA, MOHAMED, KLAUS BERBERICH, SHADY ELBASSUONI, MAYA RAMANATH, VOLKER TRESP AND GERHARD WEIKUM, 2012. Natural Language Questions for the Web of Data. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL '12*, pages 379–390. Association for Computational Linguistics, Stroudsburg, PA, USA. [Cited at pg. [11](#), [40](#), [42](#)]
- YAHYA, MOHAMED, KLAUS BERBERICH, SHADY ELBASSUONI AND GERHARD WEIKUM, 2013a. Robust Question Answering over the Web of Linked Data. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM '13*, pages 1107–1116. ACM, New York, NY, USA. ISBN 978-1-4503-2263-8. [Cited at pg. [36](#)]
- YAHYA, MOHAMED, KLAUS BERBERICH, MAYA RAMANATH AND GERHARD WEIKUM, 2013b. On the SPOT: Question Answering over Temporally Enhanced Structured Data. In *SIGIR 2013 Workshop on Time-aware Information Access*. [Cited at pg. [9](#), [36](#), [37](#), [40](#), [42](#), [80](#)]
- YANG, HUI AND TAT-SENG CHUA, 2004a. Effectiveness of web page classification on finding list answers. In MARK SANDERSON, KALERVO JÄRVELIN, JAMES ALLAN AND PETER BRUZA, editors, *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 522–523. ACM. ISBN 1-58113-881-4. [Cited at pg. [29](#), [32](#), [132](#)]
- YANG, HUI AND TAT-SENG CHUA, 2004b. Web-based List Question Answering. In *Proceedings of the 20th international conference on Computational Linguistics (COLING '04)*, page 1277. Association for Computational Linguistics, Morristown, NJ, USA. [Cited at pg. [32](#)]

REFERENCES

- YANG, HUI, HANG CUI, MSTISLAV MASLENNIKOV, LONG QIU, MIN-YEN KAN AND TAT-SENG CHUA, 2003. QUALIFIER In TREC-12 QA Main Task. In *Proceedings of the 12th Text REtrieval Conference (TREC-12)*, pages 480–488. [Cited at pg. [28](#), [32](#)]
- ZHANG, DELL AND WEE SUN LEE, 2003. A Web-based Question Answering System. In *Proceedings of The Singapore-MIT Alliance (SMA) Annual Symposium 2003*. Singapore. [Cited at pg. [9](#), [34](#), [37](#)]
- ZHENG, ZHIPING, 2002. AnswerBus Question Answering System. In *Proceedings of the Second International Conference on Human Language Technology Research, HLT '02*, pages 399–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. [Cited at pg. [9](#), [33](#), [37](#)]
- ZHOU, YAQIAN, XIAOFENG YUAN, JUNKUO CAO, XUANJING HUANG AND LIDE WU, 2006. FDUQA on TREC 2006 QA Track. In ELLEN M. VOORHEES AND LORI P. BUCKLAND, editors, *Proceedings of the Fifteenth Text REtrieval Conference, TREC 2006, Gaithersburg, Maryland, November 14-17, 2006*, volume Special Publication 500-272. National Institute of Standards and Technology (NIST). [Cited at pg. [29](#)]